



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출원 번호 : 10-2002-0053326  
Application Number PATENT-2002-0053326

출원 년 월 일 : 2002년 09월 04일  
Date of Application SEP 04, 2002

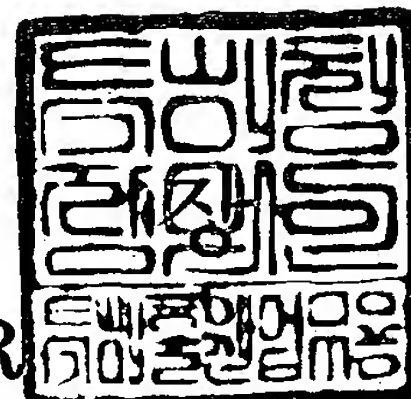
출원인 : 삼성전자 주식회사  
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



2002 년 12 월 18 일

특 허 청

COMMISSIONER



## 【서지사항】

【서류명】	특허출원서
【관리구분】	특허
【수신처】	특허청장
【참조번호】	0021
【제출일자】	2002.09.04
【국제특허분류】	H01L
【발명의 명칭】	인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card) 및 인 터럽트 발생방법
【발명의 영문명칭】	Network Interface Card for reducing the number of interrupt and method thereof
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이영필
【대리인코드】	9-1998-000334-6
【포괄위임등록번호】	1999-009556-9
【대리인】	
【성명】	정상빈
【대리인코드】	9-1998-000541-1
【포괄위임등록번호】	1999-009617-5
【발명자】	
【성명의 국문표기】	오화석
【성명의 영문표기】	OH, Hwa Seok
【주민등록번호】	720209-1000313
【우편번호】	110-010
【주소】	서울특별시 종로구 신영동 150-6 삼성중앙빌라 1-203
【국적】	KR
【취지】	특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대 리인 필 (인) 대리인 정상빈 (인)

이영

## 【수수료】

【기본출원료】	20	면	29,000	원
---------	----	---	--------	---

【가산출원료】	63	면	63,000	원
---------	----	---	--------	---

【우선권주장료】	0	건	0	원
----------	---	---	---	---

【심사청구료】	0	항	0	원
---------	---	---	---	---

【합계】	92,000	원		
------	--------	---	--	--

【첨부서류】	1. 요약서·명세서(도면)_1통			
--------	-------------------	--	--	--

**【요약서】****【요약】**

인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드 및 인터럽트 발생방법이 개시된다. 본 발명에 따른 인터럽트 발생 방법은, (a) 데이터 프레임을 수신하는 단계, (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 개수를 카운팅 하며, 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계, (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계, (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계, (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 지를 판단하고, 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 (b) 단계로 돌아가는 단계, (f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비한다. 본 발명은 발생하는 인터럽트의 수를 최소화하면서도, 한 번의 인터럽트 발생으로 많은 수의 데이터 프레임을 처리할 수 있는 장점이 있다.

**【대표도】**

**【명세서】****【발명의 명칭】**

인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card) 및 인터럽트 발생방법{Network Interface Card for reducing the number of interrupt and method thereof}

**【도면의 간단한 설명】**

본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 간단한 설명이 제공된다.

도 1은 네트워크 인터페이스 카드의 제어부의 동작 원리를 나타내는 도면이다.

도 2는 종래의 지연 시간을 가지는 인터럽트 발생 방법의 개념을 설명하는 도면이다.

도 3은 또 다른 종래의 지연 시간을 가지는 인터럽트 발생 방법의 개념을 설명하는 도면이다.

도 4는 본 발명의 제 1 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 5는 도 4의 인터럽트 발생 방법의 개념을 설명하는 도면이다.

도 6은 본 발명의 제 2 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 7은 본 발명의 제 3 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 8은 본 발명의 제 4 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 9는 본 발명의 제 5 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 10은 본 발명의 제 6 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 11은 본 발명의 제 7 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

도 12는 다른 기술적 과제를 달성하기 위한 본 발명의 제 1 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.

도 13은 다른 기술적 과제를 달성하기 위한 본 발명의 제 2 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.

도 14는 다른 기술적 과제를 달성하기 위한 본 발명의 제 3 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.

도 15는 다른 기술적 과제를 달성하기 위한 본 발명의 제 4 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

- <17>      본 발명은 네트워크 인터페이스 카드(Network Interface Card)에 관한 것으로서, 특히 이더넷(ethernet)에 사용되는 인터럽트의 발생 횟수를 최소화하는 네트워크 인터페이스 카드 및 인터럽트 발생 방법에 관한 것이다.
- <18>      현재 가장 널리 사용되고 있는 100Mbps 급 이더넷보다 10배 빠른 속도를 보장하는 기가비트 이더넷(gigabit ethernet)에 사용되는 컨트롤러(controller)의 성능을 향상시키기 위해서는 데이터의 송수신시 가장 많은 시간을 소비하는 인터럽트 전송 단계를 최적화해야한다. 즉, 동일한 양의 데이터를 처리함에 있어서 발생하는 인터럽트의 수를 줄이면 시스템의 오버헤드(overhead) 작업이 줄어들게 되므로 그만큼 컨트롤러의 성능이 향상된다.
- <19>      이더넷(Ethernet) 랜(LAN)망을 구축하기 위해서는 각 사용자의 개인용 컴퓨터 단말기(PC)에 네트워크 인터페이스 카드(Network Interface Card)를 장착하게 된다. 사용자의 단말기에 장착되는 네트워크 인터페이스 카드(NIC)는 사용자가 이더넷 랜 망을 사용할 수 있게 해주는 장치로서 제어부와 트랜시버를 구비한다. 이중 네트워크 인터페이스 카드의 성능을 결정하는 요소는 제어부 이다. 제어부는 단말기와의 인터페이스를 위한 버스 인터페이스 부와 메모리에 데이터를 기입하거나 메모리로부터 데이터를 독출하기 위한 디엠에이(DMA : Direct Memory Access) 부 및 이더넷의 프로토콜(Protocol) 처리를 위한 엠에이씨(MAC :Media Access Control) 코어를 구비한다.

- <20> 도 1은 네트워크 인터페이스 카드의 제어부의 동작 원리를 나타내는 도면이다.
- <21> 도 1을 참조하면 네트워크 인터페이스 카드(100)는 제어부(110) 및 트랜시버(120)를 구비한다. 제어부(110)는 버스 인터페이스부(111)와 디엠에이 부(113) 및 엠에이씨 코어(115)를 구비한다.
- <22> 도 1을 참조하여 네트워크 인터페이스 카드(100)가 데이터를 송수신하는 동작을 설명한다.
- <23> 송신해야할 데이터 프레임(DF)이 있는 경우, 데이터 프레임(DF)은 운영 체제의 상위 프로토콜(예컨대, TCP/IP :Transmission Control Protocol/Internet Protocol, UDP : User Datagram Protocol)등을 통하여 구동 프로그램(Device Driver)(150)으로 전달된다. 여기서, 구동 프로그램(150)은 네트워크 인터페이스 카드(100)를 구동하는 소프트웨어이다.(단계-(1))
- <24> 구동 프로그램(150)은 제어부(110)의 디엠에이(DMA)(113)가 액세스하는 메모리영역, 즉 제 1 메모리(Tx Queue)(130)로 송신해야할 데이터 프레임(DF)을 옮겨 놓는다. (단계-(2)) 그리고 구동 프로그램(150)은 제어부(110)에 송신 명령을 내린다.(단계-(3))
- <25> 송신 명령을 받은 제어부(110)는 디엠에이(113)를 이용하여 제 1 메모리(Tx Queue)(130)를 검사하고, 송신할 데이터 프레임(DF)을 제어부(110) 내부로 옮기고 (단계-(4)) 이를 처리하여 외부 네트워크(미도시)로 전송하게 된다.(단계-(5))



- <26> 외부 네트워크로 전송하기 위해서는 준비된 데이터를 이더넷 프레임(Ethernet Frame)으로 구성하는 작업이 필요하게 되는데 이러한 작업은 제어부(110) 내부의 엠에이씨 코어(MAC core)(115)에서 수행된다.
- <27> 데이터 프레임(DF)을 수신하는 경우에는 반대의 동작이 수행된다. 구동 프로그램(150)은 먼저 데이터 프레임(DF)의 수신을 위해서, 메모리 내에 수신된 데이터 프레임(DF)을 저장할 제 2 메모리(Rx Queue)(140)를 할당하고 이를 제어부(110)에 알려 준다.
- <28> 네트워크 인터페이스 카드(100) 내의 트랜시버(120)로부터 데이터 프레임(DF)이 수신되면 먼저 엠에이씨 코어(115)에서 유효한 이더넷 데이터 프레임(DF)인지를 검사하고 수신여부를 판단한다.(단계-(a))
- <29> 엠에이씨 코어(115)에서 수신여부가 결정되면 데이터 프레임(DF)은 디엠에이(113)로 전달된다. 디엠에이(113)에서는 수신된 데이터 프레임(DF)을 제 2 메모리(Rx Queue)(140)로 전송한다.(단계-(b)) 데이터 프레임(DF)이 모두 전송되면, 인터럽트(INTS)를 발생하여 수신된 데이터 프레임(DF)을 전송할 준비가 되었음을 구동 프로그램(150)으로 알리게 된다.(단계-(c))
- <30> 네트워크 인터페이스 카드(100)에서 발생된 인터럽트(INTS)는 운영 체제의 처리를 거쳐서 구동 프로그램(150)으로 전달되고(단계-(d)) 구동 프로그램(150)은 인터럽트(INTS)를 수신하면 제 2 메모리(Rx Queue)(140)를 검사하고 수신된 데이터 프레임(DF)을 찾아서(단계-(e)) 상위 프로토콜로 전송하게 된다. (단계-(f))

- <31> 이러한 동작에서 엠에이씨 코어(115)나 트랜시버(120)는 송수신되는 데이터 프레임(DF)을 저장하는 기능을 수행하지 않으므로, 송수신 성능을 결정짓는 요소는 디엠에이(113)의 작업 시간이나, 구동 프로그램(150)의 작업 시간 등이다.
- <32> 즉, 디엠에이(113)가 제 1 메모리(Tx Queue)(130)나 제 2 메모리(Rx Queue)(140)에 데이터 프레임(DF)을 기입 또는 독출하는 과정에서 불필요한 작업이 많거나 시간 지연이 길어지면 결국에는 데이터 프레임(DF)하나를 전송하는데 걸리는 시간이 늘어나게 되고 네트워크 인터페이스 카드(100)의 성능은 저하된다.
- <33> 그러나 데이터 프레임의 송수신 성능에 더욱 커다란 영향을 주는 것은 구동 프로그램(150)의 동작시간이다. 디엠에이(113)의 동작은 하드웨어에 의하여 수행되는 작업으로서 소프트웨어에 의하여 수행되는 구동 프로그램(150)의 동작에 비해서 오버헤드가 있다 하더라도 상대적으로 네트워크 인터페이스 카드(100)의 성능에 미치는 영향은 적다. 따라서 네트워크 인터페이스 카드(100)의 제어부(110)의 성능을 높이기 위해서는 구동 프로그램(150)이 최소한의 작업으로 데이터 프레임(DF)을 처리할 수 있도록 디엠에이(113)의 작업을 최적화하는 것이 필요하다.
- <34> 데이터 프레임(DF)의 수신 시에 가장 많은 시간이 소요되는 것은 인터럽트(INTS)의 전송이다. 디엠에이(113)의 동작이 끝나고 인터럽트(INTS)가 발생되면, 인터럽트(INTS)는 운영 체제에 의해서 먼저 처리된다. 운영 체제는 발생된 인터럽트(INTS)를 감지하면, 수행 중이던 동작을 멈추고 인터럽트(INTS) 처리 루틴을 수행한다. 인터럽트 처리 루틴은 발생된 인터럽트(INTS)를 처리해줄 프로그램을 찾게 되고 이때 구동 프로그램(150)의 인터럽트 처리 루틴이 실행된다. 즉, 인터럽트(INTS)의 처리 시에 수행되는 작업들은 복잡한 단계를 거치게 되고 각 단계들은 모두 운영체제를 비롯한 소프트웨어의 작업이다

따라서 많은 시간이 소비되게 되므로, 인터럽트(INTS)의 호출이 많아지면 CPU가 사용자의 작업을 처리하는데 그만큼 적은 시간을 사용하게 되므로 시스템 전체의 성능이 저하되는 문제가 있다. 따라서, 한번의 인터럽트(INTS)에 최대한 많은 데이터를 처리해주는 것이 시스템 전체의 성능을 향상시키는 방법이다.

<35> 도 2는 종래의 지연 시간을 가지는 인터럽트 발생 방법의 개념을 설명하는 도면이다.

<36> 종래의 지연 시간을 이용하는 인터럽트 발생 방법은 첫 번째 데이터 프레임(DF1)을 수신하여 인터럽트를 발생해야하는 경우, 인터럽트를 바로 발생시키는 것이 아니라 미리 정하여진 지연 시간(TD)만큼 경과된 후에 인터럽트를 발생시키는 방법이다.

<37> 먼저 데이터 프레임들(DF1 ~DF3)을 수신한다. 첫 번째 데이터 프레임(DF1)에 응답하여 일정한 지연 시간(TD)을 측정한다. 인터럽트 발생을 위해 지연되는 지연 시간(TD)은 구동 프로그램(150)이 제어할 수 있도록 네트워크 인터페이스 카드(100) 내부의 CSR 레지스터(Control and Status Register)에 저장되도록 한다. 지연 시간(TD)은 하나의 데이터 프레임(DF)을 수신하는 시간보다는 길어야 할 것이다.

<38> 첫 번째 데이터 프레임(DF1)이 수신되더라도 바로 인터럽트가 발생되지 않고, 인터럽트는 지연 시간(TD)이 소진된 후에 발생된다. 지연 시간(TD)동안에 수신된 데이터 프레임(DF1, DF2)도 디엠에이(113)에 의하여 시스템으로 전달되므로 한번의 인터럽트에 의하여 보다 많은 데이터 프레임(DF)을 전달할 수 있게 된다.

<39> 도 2에서 알 수 있는 것처럼 종래의 인터럽트 발생 방법을 사용하면 한번의 인터럽트에 2개의 데이터 프레임(DF1, DF2)을 처리할 수 있으므로 두 번째 데이터

프레임(DF2)을 위한 인터럽트 한번을 절약할 수 있다. 여기서 TP는 인터럽트가 발생되고 구동 프로그램의 인터럽트 처리 루틴이 호출되는 때까지의 지연 시간을 나타낸다.

<40> 도 3은 또 다른 종래의 지연 시간을 가지는 인터럽트 발생 방법의 개념을 설명하는 도면이다.

<41> 이 방법은 정해진 개수의 데이터 프레임(DF)이 수신되면, 지연 시간(TD)에 도달되지 아니하여도 인터럽트를 발생하도록 하는 것이다. 이때 수신되는 최대 데이터 프레임(DF)의 개수 N도 역시 CSR 레지스터(Control and Status Register)에 저장되어 제어된다

<42> 도 2에서 설명된 지연 시간(TD)에 도달되지 않으면, 데이터 프레임의 개수가 N 개 인지를 판단한다. 여기서, N은 지연 시간(TD)에 도달되기 전에 제 2 메모리(140)에 데이터 프레임(DF)이 가득 찰 수 있는 개수보다 적은 수 이어야 한다.

<43> 지연 시간(TD)에 도달되기 전에 정하여진 데이터 프레임(DF)의 개수에 도달하면 인터럽트를 발생한다. 물론, 정하여진 데이터 프레임(DF)의 개수에 도달되기 전에 지연 시간(TD)에 먼저 도달되면 역시 인터럽트를 발생한다.

<44> 지연 시간(TD)에 도달되기 전에 데이터 프레임(DF1 ~ DF4)이 정해진 개수(여기서는 4개)만큼 수신되면 인터럽트를 발생하여 수신 적체를 막을 수 있다.

<45> 그러나 데이터 프레임 사이의 시간 간격이 큰 경우 처음 수신된 데이터 프레임이 처리되는데 걸리는 시간이 길어지는 문제점이 있으며 한번의 인터럽트로 더 많은 수의 데이터 프레임을 처리하는 것이 필요하다.

**【발명이 이루고자 하는 기술적 과제】**

<46> 본 발명이 이루고자하는 기술적 과제는, 네트워크 인터페이스 카드에 서 데이터 프레임 수신하는 경우 발생하는 인터럽트의 수를 최소화하며, 한번의 인터럽트에 의하여 많은 수의 데이터 프레임을 처리하는 방법을 제공하는데 있다.

<47> 본 발명이 이루고자하는 다른 기술적 과제는, 데이터 프레임을 수신하는 경우 발생하는 인터럽트의 수를 최소화하며, 한번의 인터럽트에 의하여 많은 수의 데이터 프레임을 처리하는 네트워크 인터페이스 카드를 제공하는데 있다.

**【발명의 구성 및 작용】**

<48> 상기 기술적 과제를 달성하기 위한 본 발명의 제 1 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계 (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고 상기 수신되는 데이터 프레임의 갯수를 카운팅하는 단계 (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간에 도달하였으면 인터럽트를 발생하고 상기 제 2 지연 시간에 도달하지 않았으면 상기 (b) 단계로 돌아가는 단계 (f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간 및 상기 제 2 지연 시간을 측정하는 동작 및 상기 데이터 프레임의 개수를 카운팅하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧다.

<49>      상기 기술적 과제를 달성하기 위한 본 발명의 제 2 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계 (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 패킷 지연 시간을 측정하는 단계 (c) 상기 패킷 지연 시간에 도달했는지를 판단하고, 상기 패킷 지연 시간에 도달했으면 인터럽트를 발생하고 상기 패킷 지연 시간에 도달하지 않았으면 상기 (b) 단계로 돌아가는 단계 (e) 상기 발생된 인터럽트에 응답하여 상기 패킷 지연 시간을 측정하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (f) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 한다. 상기 패킷 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값이다.

<50>      상기 기술적 과제를 달성하기 위한 본 발명의 제 3 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface

Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계 (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 갯수를 카운팅 하며, 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하는 단계 (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고, 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 (b) 단계로 돌아가는 단계 (f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.

<51>        상기 기술적 과제를 달성하기 위한 본 발명의 제 4 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계 (b) 상기 수신되는 데이터 프레임의 타입 필드(type field)의 값을 인식하는 단계 (c) 상



기 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b)단계로 돌아가는 단계 (d) 상기 발생된 인터럽트에 응답하여 상기 타입 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (e) 새로운 데이터 프레임을 수신하고 상기(b)단계로 돌아가는 단계를 구비하는 것을 특징으로 한다.

<52>      상기 기술적 과제를 달성하기 위한 본 발명의 제 5 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계, (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 개수를 카운팅 하며, 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계, (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계, (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계, (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 지를 판단하고, 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 (b) 단계로 돌아가는 단계, (f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 한다.



- <53>      상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.
- <54>      상기 기술적 과제를 달성하기 위한 본 발명의 제 6 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, (a) 데이터 프레임을 수신하는 단계, (b) 상기 수신되는 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계, (c) 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는지를 판단하고 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b)단계로 돌아가는 단계, (d) 상기 발생된 인터럽트에 응답하여 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (e) 새로운 데이터 프레임을 수신하고 상기(b)단계로 돌아가는 단계를 구비하는 것을 특징으로 한다.
- <55>      상기 기술적 과제를 달성하기 위한 본 발명의 제 7 실시예에 따른 인터럽트 발생 방법은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서 (a) 데이터 프레임을 수신하는 단계 (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 갯수를 카운팅 하고 상기 수신되는 데이터 프레임의 타입 필드(type field)의 값을 인식하며, 상기 수신되는 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계 (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가

N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간에 도달하였으면 인터럽트를 발생하는 단계, (f) 상기 제 2 지연 시간에 도달하지 않았으면, 상기 타입 필드의 값이 미리 지정된 타입 값과 일치하는지를 판단하고 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하는 단계, (g) 상기 미리 지정된 타입 값과 일치하지 않으면, 상기 수신되는 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는지를 판단하고 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b) 단계로 돌아가는 단계 (h) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간 및 상기 제 2 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작, 상기 타입 필드의 값을 인식하는 동작 및 상기 프로토콜 타입 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 및 (i) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧다.

<56> 다른 기술적 과제를 달성하기 위한 본 발명의 제 1 실시예에 따른 네트워크 인터페이스 카드는, 제 1 지연 시간 측정 회로, 제 2 지연 시간 측정 회로, 데이터 프레임 카운팅 회로, 판단 회로 및 인터럽트 발생 회로를 구비하는 것을 특징으로 한다.

<57> 제 1 지연 시간 측정 회로는 수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지한다.

- <58> 제 2 지연 시간 측정 회로는 상기 수신되는 데이터 프레임에 응답하여 일정한 제 2 지연 시간을 측정하고 상기 인터럽트에 응답하여 상기 제 2 지연 시간의 측정을 정지한다.
- <59> 데이터 프레임 카운팅 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지한다.
- <60> 판단 회로는 상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 제 2 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 2 지연 시간 측정 회로가 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생한다.
- <61> 인터럽트 발생 회로는 상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한다. 값으로서, 상기 제 1 지연 시간보다 짧다.
- <62> 상기 네트워크 인터페이스 카드(Network Interface Card)는 상기 데이터 프레임을 수신하여 상기 제 1 지연 시간 측정 회로, 상기 제 2 지연 시간 측정 회로 및 상기 데이터 프레임 카운팅 회로로 인가하는 수신 회로 및 상기 인터럽트를 수신하여 전송하는 전송 회로를 더 구비한다.

- <63> 다른 기술적 과제를 달성하기 위한 본 발명의 제 2 실시예에 따른 네트워크 인터페이스 카드는, 제 1 지연 시간 측정 회로, 타입 인식 회로, 데이터 프레임 카운팅 회로, 판단 회로 및 인터럽트 발생 회로를 구비하는 것을 특징으로 한다.
- <64> 제 1 지연 시간 측정 회로는 수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지한다.
- <65> 타입 인식 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하고 상기 인터럽트에 응답하여 상기 타입 필드의 값의 인식을 정지한다.
- <66> 데이터 프레임 카운팅 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지한다.
- <67> 판단 회로는 상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 타입 인식 회로의 출력 신호에 응답하여 상기 타입 인식 회로가 미리 지정된 타입 값과 일치하는지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생한다.
- <68> 인터럽트 발생 회로는 상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 한다.

- <69> 다른 기술적 과제를 달성하기 위한 본 발명의 제 3 실시예에 따른 네트워크 인터페이스 카드는 제 1 지연 시간 측정 회로, 프로토콜 인식 회로, 데이터 프레임 카운팅 회로, 판단 회로 및 인터럽트 발생 회로를 구비하는 것을 특징으로 한다.
- <70> 제 1 지연 시간 측정 회로는 수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지한다.
- <71> 프로토콜 인식 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 상기 인터럽트에 응답하여 상기 프로토콜 필드 값의 인식을 정지한다.
- <72> 데이터 프레임 카운팅 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅 하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지한다.
- <73> 판단 회로는 상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 프로토콜 인식 회로의 출력 신호에 응답하여 상기 프로토콜 인식 회로가 인식한 프로토콜 값이 미리 지정된 프로토콜 값과 일치하는 지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생한다.
- <74> 인터럽트 발생 회로는 상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 한다.

- <75> 다른 기술적 과제를 달성하기 위한 본 발명의 제 4 실시예에 따른 네트워크 인터페이스 카드는
- <76> 제 1 지연 시간 측정 회로, 제 2 지연 시간 측정 회로, 타입 인식 회로, 데이터 프레임 카운팅 회로, 프로토콜 인식 회로, 판단 회로 및 인터럽트 발생 회로를 구비하는 것을 특징으로 한다.
- <77> 제 1 지연 시간 측정 회로는 수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지한다.
- <78> 제 2 지연 시간 측정 회로는 상기 수신되는 데이터 프레임에 응답하여 일정한 제 2 지연 시간을 측정하고 상기 인터럽트에 응답하여 상기 제 2 지연 시간의 측정을 정지한다.
- <79> 타입 인식 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하고 상기 인터럽트에 응답하여 상기 타입 필드의 값의 인식을 정지한다.
- <80> 데이터 프레임 카운팅 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅 하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지한다.
- <81> 프로토콜 인식 회로는 상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 상기 인터럽트에 응답하여 상기 프로토콜 필드 값의 인식을 정지한다.

- <82> 판단 회로는 상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 2 지연 시간 측정 회로가 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 타이밍 인식 회로의 출력 신호에 응답하여 상기 타이밍 인식 회로가 인식하는 타이밍 값이 미리 지정된 타이밍 값과 일치하는지를 판단하며, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 프로토콜 인식 회로의 출력 신호에 응답하여 상기 프로토콜 인식 회로가 인식하는 프로토콜 값이 미리 지정된 프로토콜 값과 일치하는지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생한다.
- <83> 인터럽트 발생 회로는 상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧다.
- <84> 상기 네트워크 인터페이스 카드(Network Interface Card)는 상기 데이터 프레임을 수신하여 상기 제 1 지연 시간 측정 회로, 상기 제 2 지연 시간 측정 회로, 상기 데이터 프레임 카운팅 회로 및 상기 프로토콜 인식 회로로 인가하는 수신 회로 및 상기 인터럽트를 수신하여 전송하는 전송 회로를 더 구비한다.
- <85> 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시예를 예시하는 첨부 도면 및 도면에 기재된 내용을 참조하여야 한다.



- <86>        이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시예를 설명함으로써, 본 발명을 상세히 설명한다. 각 도면에 제시된 동일한 참조부호는 동일한 부재를 나타낸다.
- <87>        도 4는 본 발명의 제 1 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.
- <88>        도 5는 도 4의 인터럽트 발생 방법의 개념을 설명하는 도면이다.
- <89>        도 4 및 도 5를 참조하면, 본 발명의 제 1 실시예에 따른 인터럽트 발생 방법(400)은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(410 단계), 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고 수신되는 데이터 프레임의 개수를 카운팅하는 단계(420 단계), 제 1 지연 시간에 도달했는지를 판단하고, 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계(430 및 460 단계), 제 1 지연 시간에 도달하지 않았으면 데이터 프레임의 개수가 N 개인지를 카운팅하고 N 개이면 인터럽트를 발생하는 단계(제 440 및 제 460 단계), 데이터 프레임의 개수가 N 개가 아니면 제 2 지연 시간에 도달하였는지를 판단하고, 제 2 지연 시간에 도달하였으면 인터럽트를 발생하고 제 2 지연 시간에 도달하지 않았으면 제420 단계로 돌아가는 단계(450 및 460 단계), 발생한 인터럽트에 응답하여 제 1 지연 시간 및 제 2 지연 시간을 측정하는 동작 및 데이터 프레임의 개수를 카운팅하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계(470 단계) 및 새로운 데이터 프레임을 수신하고 제 420 단계로 돌아가는 단계(480 단계)를 구비하는 것을 특징으로 한다.



- <90>      상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧다.
- <91>      이하 도 4 및 도 5를 참조하여 본 발명의 제 1 실시예에 따른 인터럽트 발생 방법이 상세히 설명된다.
- <92>      제 1 실시예에 따른 인터럽트 발생 방법(400)은 수신되는 데이터 프레임(DF1, DF2) 사이의 간격이 크고 제 1 지연 시간(TD1)의 길이가 길게 설정 된다면 첫 번째 데이터 프레임(DF1)이 처리될 때까지 걸리는 시간이 상대적으로 길게 된다. 이러한 문제를 해결 하기 위해서, 데이터 프레임(DF1, DF2)사이의 수신 시간이 일정 값을 넘어간다면 인터럽트를 발생시켜 이미 수신된 데이터 프레임(DF1)을 처리하는 것이다.
- <93>      데이터 프레임(DF1)을 수신하여 일정한 제 1 지연 시간(TD1)을 측정한다.(410 내지 430 단계) 제 1 지연 시간(TD1)은 첫 번째 데이터 프레임(DF1)을 수신한 때부터 측정되는 일정한 시간이다. 제 1 지연 시간(TD1)에 도달되면 인터럽트를 발생한다. 제 1 지연 시간(TD1)에 도달되지 않으면, 데이터 프레임(DF)의 개수가 N 개인지를 판단한다.(제 440 단계) 데이터 프레임(DF)의 개수가 N 개라면 데이터 프레임(DF)이 미리 정하여진 수 만큼 누적된 것이므로 데이터의 처리를 위하여 인터럽트가 발생된다. 데이터 프레임(DF)의 개수가 N 개가 아니라면 제 2 지연 시간(TD2)에 도달했는지를 판단한다.(450 단계) 제 2 지연 시간(TD2)은 수신되는 데이터 프레임들(DF1, DF2) 사이의 시간 간격을 측정한 값으로서, 제 1 지연 시간(TD1)보다 짧다.
- <94>      첫 번째 데이터 프레임(DF1)을 수신한 후 제 2 지연 시간(TD2)이 경과될 동안 두 번째 데이터 프레임(DF2)이 수신되지 않으면 제 2 지연 시간(TD2)이 경과되는 순간에 인

터럽트를 발생한다. 물론 첫 번째 데이터 프레임(DF1)을 수신한 후 제 2 지연 시간(TD2)이 경과되기 전에 두 번째 데이터 프레임(DF2)이 수신되면 인터럽트는 발생되지 않으며 두 번째 데이터 프레임(DF2)을 수신한 때부터 다시 제 2 지연 시간(TD2)을 측정할 것이다. 제 2 지연 시간(TD2)이 경과되기 전에 계속하여 데이터 프레임(DF)이 수신되면, 즉, 데이터 프레임들간의 시간 간격이 제 2 지연 시간(TD2)보다 짧다면, 제 1 지연 시간(TD1)이 경과되거나 수신되는 데이터 프레임(DF)의 개수가 N 개가 누적되는 순간에 인터럽트가 발생할 것이다.

<95> 제 2 지연 시간(TD2)을 측정하는 것은 타이머를 이용할 수 있으나 반드시 타이머에 한정되는 것은 아니다.

<96> 인터럽트가 발생되면, 발생한 인터럽트에 응답하여 제 1 및 제 2 지연 시간(TD1, TD2)을 측정하는 동작 및 데이터 프레임(DF)을 카운팅 하는 동작을 정지시키고 그동안 수신된 데이터 프레임(DF1)을 상위 프로토콜로 전송한다.(470 단계) 또한 수신된 데이터 프레임(DF1)을 모두 전송하면, 다시 새로운 데이터 프레임(DF2)을 수신하여 제 1 및 제 2 지연 시간(TD1, TD2)을 측정하거나 데이터 프레임의 개수를 카운팅 하는 제 420 단계의 동작을 시작한다.(480 단계)

<97> 도 5를 보면, 제 2 지연 시간(TD2)에 도달되기 전에 새로운 데이터 프레임(DF2)이 수신되지 않으므로 제 2 지연 시간(TD2) 경과 후 인터럽트가 발생된다.

<98> 도 6은 본 발명의 제 2 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

- <99>       도 6을 참조하면, 본 발명의 제 2 실시예에 따른 인터럽트 발생 방법(600)은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(610 단계), 수신되는 데이터 프레임에 응답하여 미리 정하여진 패킷 지연시간을 측정하는 단계(620 단계), 패킷 지연 시간에 도달했는지를 판단하는 단계(630 단계), 패킷 지연 시간에 도달했으면 인터럽트를 발생하는 단계(640 단계), 발생한 인터럽트에 응답하여 패킷 지연 시간을 측정하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계(650 단계) 및 새로운 데이터 프레임을 수신하고 제 620 단계로 돌아가는 단계(660 단계)를 구비한다.
- <100>       패킷 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값이다.
- <101>       이하 도 6을 참조하여 본 발명의 제 2 실시예에 따른 인터럽트 발생 방법이 상세히 설명된다.
- <102>       제 2 실시예에 따른 인터럽트 발생 방법(600)은 제 1 실시예에 따른 인터럽트 발생 방법(400)중 제 2 지연 시간을 이용하는 방법이다.
- <103>       즉, 패킷 지연 시간은 제 1 실시예의 제 2 지연 시간(TD2)과 동일한 개념이다.
- <104>       데이터 프레임을 수신하여 일정한 패킷 지연 시간을 측정한다.(610 내지 620 단계) 패킷 지연 시간에 도달했는지를 판단한다.(630 단계) 패킷 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값이다.
- <105>       첫 번째 데이터 프레임을 수신한 후 패킷 지연 시간이 경과될 동안 두 번째 데이터 프레임이 수신되지 않으면 패킷 지연 시간이 경과되는 순간에 인터럽트를 발생한다. 물론 첫 번째 데이터 프레임을 수신한 후 패킷 지연 시간이 경과되기 전에 두 번째 데이터

프레임이 수신되면 인터럽트는 발생되지 않으며 두 번째 데이터 프레임을 수신한 때부터 다시 패킷 지연 시간을 측정한다.

<106> 패킷 지연 시간을 측정하는 것은 타이머를 이용할 수 있으나 반드시 타이머에 한정되는 것은 아니다.

<107> 인터럽트가 발생되면, 발생한 인터럽트에 응답하여 패킷 지연 시간을 측정하는 동작을 정지시키고 그동안 수신된 데이터 프레임을 상위 프로토콜로 전송한다.(650 단계)  
또한 수신된 데이터 프레임을 모두 전송하면, 다시 새로운 데이터 프레임을 수신하여 패킷 지연 시간을 측정하는 제 620 단계의 동작을 시작한다.(660 단계)

<108> 도 7은 본 발명의 제 3 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

<109> 도 7을 참조하면, 본 발명의 제 3 실시예에 따른 인터럽트 발생 방법(700)은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(710 단계), 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고 수신되는 데이터 프레임의 개수를 카운팅 하며 데이터 프레임의 타입 필드(type field)의 값을 인식하는 단계(720 단계), 제 1 지연 시간에 도달했는지를 판단하고, 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계(730 및 760 단계), 제 1 지연 시간에 도달하지 않았으면 데이터 프레임의 개수가 N 개인지를 카운팅 하고 N 개이면 인터럽트를 발생하는 단계(제 740 및 제 760 단계), 데이터 프레임의 개수가 N 개가 아니면 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고, 일치하면 인터럽트를 발생하고 일치하지 않았으면 제 720 단계로 돌아가는 단계(750 및 760 단계), 발생한 인터럽트에 응답하여 제 1 지연 시간을 측

정하는 동작, 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 데이터 프레임의 타입 필드의 타입 값을 인식하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계 (770 단계) 및 새로운 데이터 프레임을 수신하고 제 720 단계로 돌아가는 단계(780 단계)를 구비한다.

<110>       상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.

<111>       이하 도 7을 참조하여 본 발명의 제 3 실시예에 따른 인터럽트 발생 방법이 상세히 설명된다.

<112>       수신되는 데이터 프레임들 중에는 정해진 시간 내에 반드시 처리되어야 하는 데이터를 포함하는 데이터 프레임들이 있다. 따라서 이러한 데이터 프레임들을 미리 지정하고 이러한 데이터 프레임들이 수신되는 경우 이들 데이터 프레임들을 인식하여 인터럽트를 발생하는 것이다.

<113>       데이터 프레임에는 목적 주소(destination address), 소스 주소(source address), 타입(type), 데이터(data), CRC(cyclic redundancy check)등을 저장하는 필드(field)들이 존재한다. 그 중 타입(type)필드에는 데이터 프레임에 저장되는 데이터의 타입이 어떠한 것인지에 대한 정보가 저장된다.

<114>       일정한 시간 내에 반드시 처리되어야 하는 데이터의 타입을 미리 지정하고, 수신되는 데이터 프레임의 타입 필드를 검색하여 검색된 타입 값이 미리 지정한 타입 값과 동일하면 인터럽트를 발생한다.

<115> 이때 미리 지정된 타임 값은 네트워크 인터페이스 카드 내부의 CSR 레지스터 (Control and Status Register)에 저장될 수 있다.

<116> 데이터 프레임을 수신하여 일정한 제 1 지연 시간을 측정한다. (710 및 720 단계)  
제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.  
제 1 지연 시간에 도달했는지를 판단하고 제 1 지연 시간에 도달되면 인터럽트를 발생한다. (730 및 760 단계) 제 1 지연 시간에 도달되지 않으면, 데이터 프레임의 개수가 N 개 인지를 판단한다. (제 740 단계) 데이터 프레임의 개수가 N 개라면 데이터 프레임이 미리 정하여진 수만큼 누적된 것이므로 데이터의 처리를 위하여 인터럽트가 발생된다. 데이터 프레임의 개수가 N 개가 아니라면 타임 필드의 값이 미리 지정된 타임 값과 일치하는지를 판단한다. (750 단계) 데이터 프레임에서 타임 필드의 위치는 정해져 있다. 보통, 목적 주소(destination address)와 소스 주소(source address)가 12 바이트(byte)의 길이를 가진다. 그리고 타임 필드는 2 바이트 길이를 가진다. 따라서 수신되는 데이터 프레임 중 앞에서 12 바이트 다음의 2 바이트를 읽는다면 그 값이 타임 필드의 값이다. 이 값과 미리 지정된 타임 값이 일치되는지를 판단하는 것이다.

<117> 수신된 데이터 프레임의 타임 필드의 값이 미리 지정된 타임 값과 일치하면 인터럽트를 발생한다. 미리 지정된 타임 값과 일치되는 타임 필드의 값을 가지는 데이터 프레임이 수신되지 않는다면 제 1 지연 시간이 경과되거나 수신되는 데이터 프레임의 개수가 N 개가 누적되는 순간에 인터럽트가 발생될 것이다.

<118> 타임 필드의 값을 인식하기 위해서는 별도의 타임 인식 회로가 필요하다. 타임 인식 회로는 수신되는 데이터 프레임에서 타임 필드의 위치를 인식하고, 타임 필드에 저장

된 값과 미리 지정된 타입 값이 일치되는 지 여부를 판단한다. 타입 인식 회로는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이다.

<119> 인터럽트가 발생되면, 발생한 인터럽트에 응답하여 제 1 지연 시간을 측정하는 동작, 데이터 프레임의 개수를 카운팅 하는 동작 및 타입 필드의 값을 인식하는 동작을 정지시키고 그동안 수신된 데이터 프레임을 상위 프로토콜로 전송한다.(770 단계) 또한 수신된 데이터 프레임을 모두 전송하면, 다시 새로운 데이터 프레임을 수신하여 제 1 지연 시간을 측정하거나 데이터 프레임의 개수를 카운팅 하거나 타입 필드의 값을 인식하는 제 720 단계의 동작을 시작한다.(780 단계)

<120> 도 8은 본 발명의 제 4 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

<121> 도 8을 참조하면, 본 발명의 제 4 실시예에 따른 인터럽트 발생 방법(800)은

<122> 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(810 단계), 데이터 프레임의 타입 필드의 값을 인식하는 단계(820 단계), 데이터 프레임의 타입 필드 값이 미리 지정된 타입 값과 일치하는 지를 판단하는 단계(830 단계), 데이터 프레임의 타입 필드 값이 미리 지정된 타입 값과 일치하면 인터럽트를 발생하는 단계(840 단계), 발생한 인터럽트에 응답하여 데이터 프레임의 타입 필드의 타입 값을 인식하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계(850 단계) 및 새로운 데이터 프레임을 수신하고 제 820 단계로 돌아가는 단계(860 단계)를 구비한다.



- <123> 이하 도 8을 참조하여 본 발명의 제 4 실시예에 따른 인터럽트 발생방법이 상세히 설명된다.
- <124> 제 4 실시예에 따른 인터럽트 발생 방법(800)은 제 3 실시예에 따른 인터럽트 발생 방법(700)중 데이터 프레임의 타입 필드의 값의 인식 방법을 이용하는 방법이다.
- <125> 수신되는 데이터 프레임들 중에는 정해진 시간 내에 반드시 처리되어야 하는 데이터를 포함하는 데이터 프레임들이 있다. 따라서 이러한 데이터 프레임들을 미리 지정하고 이러한 데이터 프레임들이 수신되는 경우 이들 데이터 프레임들을 인식하여 인터럽트를 발생하는 것이다.
- <126> 일정한 시간 내에 반드시 처리되어야 하는 데이터의 타입을 미리 지정하고, 수신되는 데이터 프레임의 타입 필드를 검색하여 검색된 타입 값이 미리 지정한 타입 값과 동일하면 인터럽트를 발생한다.
- <127> 이때 미리 지정된 타입 값은 네트워크 인터페이스 카드 내부의 CSR 레지스터(Control and Status Register)에 저장될 수 있다.
- <128> 데이터를 프레임 수신한다.(810 단계) 데이터 프레임의 타입 필드의 값을 인식한다.(820 단계) 데이터 프레임의 타입 필드의 값이 미리 지정된 타입 값과 일치하는지를 판단한다.(830 단계) 데이터 프레임에서 타입 필드의 위치는 정해져 있다. 보통, 목적 주소(destination address)와 소스 주소(source address)가 12 바이트(byte)의 길이를 가진다. 그리고 타입 필드는 2 바이트 길이를 가진다. 따라서 수신되는 데이터 프레임 중 앞에서 12 바이트 다음의 2 바이트를 읽는다면 그 값이 타입 필드의 값이다. 이 값과 미리 지정된 타입 값이 일치되는지를 판단하는 것이다.



- <129> 수신된 데이터 프레임의 타입 필드의 값이 미리 지정된 타입 값과 일치하면 인터럽트를 발생한다.(840 단계)
- <130> 타입 필드의 값을 인식하기 위해서는 별도의 타입 인식 회로가 필요하다. 타입 인식 회로는 수신되는 데이터 프레임에서 타입 필드의 위치를 인식하고, 타입 필드에 저장된 값과 미리 지정된 타입 값이 일치되는 지 여부를 판단한다. 타입 인식 회로는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이다.
- <131> 인터럽트가 발생되면, 발생한 인터럽트에 응답하여 타입 필드의 값을 인식하는 동작을 정지시키고 그동안 수신된 데이터 프레임을 상위 프로토콜로 전송한다.(850 단계)  
또한 수신된 데이터 프레임을 모두 전송하면, 다시 새로운 데이터 프레임을 수신하여 타입 필드의 값을 인식하는 제 820 단계의 동작을 시작한다.(860 단계)
- <132> 도 9는 본 발명의 제 5 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.
- <133> 도 9를 참고하면, 본 발명의 제 5 실시예에 따른 인터럽트 발생 방법(900)은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(910), 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 개수를 카운팅 하며, 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계(920), 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계(930단계), 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계(940단계), 상기 데

이더 프레임의 개수가 N 개가 아니면, 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 지를 판단하고, 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 920 단계로 돌아가는 단계(950 및 960 단계), 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계(970 단계) 및 새로운 데이터 프레임을 수신하고 상기 920 단계로 돌아가는 단계(980 단계)를 구비한다. 상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.

<134> 이하 도 9를 참조하여 본 발명의 제 5 실시예에 따른 인터럽트 발생 방법이 상세히 설명된다.

<135> 데이터 프레임에는 목적 주소(destination address), 소스 주소(source address), 타입(type), 데이터(data), CRC(cyclic redundancy check)등을 저장하는 필드(field)들이 존재한다는 것은 앞서 설명되었다.

<136> 그 중 데이터 필드에는 이더넷(ethernet)보다 상위 계층에 존재하는 프로토콜에서 생성된 패킷이 위치하게 된다. 이더넷(ethernet)보다 상위 계층에 존재하는 프로토콜중 대표적인 것으로서 인터넷 프로토콜(IP : Internet Protocol ;이하 IP 라고 한다.), ICMP(Internet Control Message Protocol) 등이 있다. 특히 IP가 이더넷의 상위 계층으로 가장 많이 사용되는 프로토콜이다.

<137> 수신되는 데이터 프레임의 데이터 필드에 존재하며, IP에서 생성되는 IP 패킷은 IP 패킷 헤더와 IP 패킷 데이터의 두 영역으로 나뉘어 진다. IP 패킷 헤더에는 IP 패킷 데이터에 저장된 프로토콜 데이터를 전송하기 위한 제어 정보들이 저장되어 있다.

- <138> IP 패킷 데이터에는 IP 보다 상위 계층인 전송 제어 프로토콜(TCP :Transmission Control Protocol; 이하 TCP라고 한다.)나 UDP(User Datagram Protocol) 같은 프로토콜 데이터가 저장된다.
- <139> IP 패킷 헤더에 저장된 여러 종류의 제어 정보들 중 IP 패킷 데이터에 저장된 프로토콜 데이터의 종류를 나타내는 제어 정보는 프로토콜 필드에 저장되어 있다.
- <140> 제 5 실시예에 따른 인터럽트 발생 방법(900)은 수신되는 데이터 프레임의 데이터 필드에 존재하는 패킷 헤더의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 동일하다면 인터럽트를 발생하는 방법이다. 미리 지정된 프로토콜 필드 값은 긴급히 인터럽트를 발생시킬 필요성에 의하여 사용자가 지정한다.
- <141> 이때 미리 지정된 프로토콜 필드의 값은 네트워크 인터페이스 카드 내부의 CSR 레지스터(Control and Status Register)에 저장될 수 있다.
- <142> 제 5 실시예에서는 데이터 프레임에 IP 패킷이 존재하는 경우에 대하여 설명하고 있으나 IP 패킷에 한정되는 것은 아니다.
- <143> 데이터 프레임을 수신하여 일정한 제 1 지연 시간을 측정한다.(910 및 920 단계)  
제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다.  
제 1 지연 시간에 도달했는지를 판단하고 제 1 지연 시간에 도달되면 인터럽트를 발생한다.(930 및 960 단계)
- <144> 제 1 지연 시간에 도달되지 않으면, 데이터 프레임의 개수가 N 개인지를 판단한다.(제 940 단계) 데이터 프레임의 개수가 N 개라면 데이터 프레임이 미리 정해진 수만큼 누적된 것이므로 데이터의 처리를 위하여 인터럽트가 발생된다.

- <145> 데이터 프레임의 개수가 N 개가 아니라면 프로토콜 필드의 값이 미리 지정된 프로토콜 필드 값과 일치하는지를 판단한다. (950 단계) 프로토콜 필드는 데이터 프레임의 데이터 필드에 존재한다. 보통, 목적 주소(destination address)와 소스 주소(source address)가 12 바이트(byte)의 길이를 가진다. 그리고 타입 필드는 2 바이트 길이를 가진다. 따라서 수신되는 데이터 프레임 중 최상위 비트에서 15 번째 바이트부터 데이터 필드에 해당한다. 데이터 필드 내부의 패킷 헤더 부분에 프로토콜 필드가 존재한다. 프로토콜 필드의 값과 미리 지정된 프로토콜 필드 값이 일치되는지를 판단하는 것이다.
- <146> 수신된 데이터 프레임의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드 값과 일치하면 인터럽트가 발생된다. 미리 지정된 프로토콜 필드 값과 일치되는 프로토콜 필드의 값을 가지는 데이터 프레임이 수신되지 않는다면 제 1 지연 시간이 경과되거나 수신되는 데이터 프레임이 N 개가 누적되는 순간에 인터럽트가 발생될 것이다.
- <147> 프로토콜 필드의 값을 인식하기 위해서는 별도의 IP 패킷 인식 회로 및 프로토콜 필드의 값을 인식하기 위한 프로토콜 필드 추출 회로가 필요하다. IP 패킷 인식 회로는 수신되는 데이터 프레임에 IP 패킷이 존재하는지 여부를 판단한다.
- <148> 프로토콜 필드 추출 회로는 IP 패킷이 존재하는 경우 프로토콜 필드의 값을 추출하여 인식한다. IP 패킷 인식 회로 및 프로토콜 필드 추출 회로는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이다.
- <149> 인터럽트가 발생되면, 발생한 인터럽트에 응답하여 제 1 지연 시간을 측정하는 동작, 데이터 프레임의 개수를 카운팅 하는 동작 및 프로토콜 필드의 값을 인식하는 동작을 정지시키고 그동안 수신된 데이터 프레임을 상위 프로토콜로 전송한다. (970 단계) 또한 수신된 데이터 프레임을 모두 전송하면, 다시 새로운 데이터 프레임을 수신하여 제 1

지연 시간을 측정하거나 데이터 프레임의 개수를 카운팅 하거나 프로토콜 필드의 값을 인식하는 제 920 단계의 동작을 시작한다. (980 단계)

<150> 도 10은 본 발명의 제 6 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

<151> 도 10을 참조하면, 본 발명의 제 6 실시예에 따른 인터럽트 발생 방법(1000)은 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(1010 단계), 데이터 프레임의 프로토콜 필드의 값을 인식하는 단계(1020 단계), 데이터 프레임의 프로토콜 필드 값이 미리 지정된 프로토콜 필드 값과 일치하는 지를 판단하는 단계(1030 단계), 데이터 프레임의 프로토콜 필드 값이 미리 지정된 프로토콜 필드 값과 일치하면 인터럽트를 발생하는 단계(1040 단계), 발생한 인터럽트에 응답하여 데이터 프레임의 프로토콜 필드의 값을 인식하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계(1050 단계) 및 새로운 데이터 프레임을 수신하고 제 1020 단계로 돌아가는 단계(1060 단계)를 구비한다.

<152> 이하 도 10을 참조하여 본 발명의 제 6 실시예에 따른 인터럽트 발생방법이 상세히 설명된다.

<153> 제 6 실시예에 따른 인터럽트 발생 방법(1000)은 제 5 실시예에 따른 인터럽트 발생 방법(900)중 데이터 프레임의 프로토콜 필드의 값의 인식 방법을 이용하는 방법이다.

- <154>      사용자가 긴급히 인터럽트를 발생시킬 필요성에 의하여 프로토콜 필드의 값을 미리 지정하고, 이러한 프로토콜 필드의 값을 포함하는 데이터 프레임들이 수신되는 경우 상기 데이터 프레임들을 인식하여 인터럽트를 발생하는 것이다.
- <155>      데이터 프레임을 수신한다.(1010 단계) 데이터 프레임의 프로토콜 필드의 값을 인식한다.(1020 단계) 프로토콜 필드는 데이터 프레임의 데이터 필드 내부의 패킷 헤더에 존재한다. 데이터 프레임의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드 값과 일치하는지를 판단한다.(1030 단계) 수신된 데이터 프레임의 타입 필드의 값이 미리 지정된 타입 값과 일치하면 인터럽트를 발생한다.(840 단계)
- <156>      프로토콜 필드의 값을 인식하기 위해서는 별도의 패킷 인식 회로 및 프로토콜 필드의 값을 인식하기 위한 프로토콜 필드 추출 회로가 필요하다. IP 패킷 인식 회로는 수신되는 데이터 프레임에 IP 패킷이 존재하는지 여부를 판단한다.
- <157>      프로토콜 필드 추출 회로는 IP 패킷이 존재하는 경우 프로토콜 필드의 값을 추출하여 인식한다. IP 패킷 인식 회로 및 프로토콜 필드 추출 회로는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이다.
- <158>      인터럽트가 발생되면, 발생된 인터럽트에 응답하여 프로토콜 필드의 값을 인식하는 동작을 정지시키고 그동안 수신된 데이터 프레임을 상위 프로토콜로 전송한다.(1050 단계) 또한 수신된 데이터 프레임을 모두 전송하면, 다시 새로운 데이터 프레임을 수신하여 프로토콜 필드의 값을 인식하는 제 1020 단계의 동작을 시작한다.(1060 단계)
- <159>      도 11은 본 발명의 제 7 실시예에 따른 인터럽트 발생 방법을 나타내는 플로우 차트이다.

<160> 도 11을 참조하면, 본 발명의 제 7 실시예에 따른 인터럽트 발생 방법(1100)은, 데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서, 데이터 프레임을 수신하는 단계(1100 단계), 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고 수신되는 데이터 프레임의 개수를 카운팅 하고 데이터 프레임의 타입 필드(type field)의 값을 인식하며 데이터 프레임의 프로토콜 필드의 값을 인식하는 단계(1120 단계), 제 1 지연 시간에 도달했는지를 판단하고, 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계(1130 및 1170 단계), 제 1 지연 시간에 도달하지 않았으면 데이터 프레임의 개수가 N 개인지를 카운팅 하고 N 개이면 인터럽트를 발생하는 단계(제 1140 및 제 1170 단계), 데이터 프레임의 개수가 N 개가 아니면 제 2 지연 시간에 도달했는지를 판단하고, 제 2 지연 시간에 도달했으면 인터럽트를 발생하는 단계(1150 및 1170 단계), 제 2 지연 시간에 도달하지 않았으면 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고, 일치하면 인터럽트를 발생하는 단계(1160 및 1170 단계), 타입 필드의 값이 미리 지정된 타입 값과 일치하지 않으면 데이터 프레임의 프로토콜 필드의 값이 미리 지정한 프로토콜 필드의 값과 일치하는 지를 판단하고 일치하면 인터럽트를 발생하고 일치하지 않으면 제 1120 단계로 돌아가는 단계(1165 및 1170 단계), 발생한 인터럽트에 응답하여 제 1 지연 시간 및 제 2 지연 시간을 측정하는 동작, 데이터 프레임의 개수를 카운팅하는 동작, 상기 데이터 프레임의 타입 필드의 타입 값을 인식하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 수신된 데이터 프레임을 전송하는 단계(1180 단계) 및 새로운 데이터 프레임을 수신하고 제 1120 단계로 돌아가는 단계(1190 단계)를 구비한다



- <161>      상기 제 1 지연 시간은 첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간이다. 상기 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧다.
- <162>      이하 도 11을 참조하여 본 발명의 제 7 실시예에 따른 인터럽트 발생 방법이 상세히 설명된다.
- <163>      제 7 실시예에 따른 인터럽트 발생 방법(1100)은 제 1 실시예에 따른 인터럽트 발생 방법(400)과 제 3 실시예에 따른 인터럽트 발생 방법(700) 및 제 5 실시예에 따른 인터럽트 발생 방법(900)을 합하여 놓은 방법이다.
- <164>      즉, 수신되는 데이터 프레임 사이의 시간 간격이 일정한 시간 이상이 되면 데이터 프레임의 적체를 방지하기 위하여 인터럽트를 발생하는 방법과 정해진 시간 내에 처리되어야 하는 데이터의 타입을 미리 지정하고 이러한 데이터를 가지는 데이터 프레임이 수신되는 지를 인식하여 인터럽트를 발생하는 방법 및 데이터 프레임의 데이터 필드 내부에 존재하는 프로토콜 필드의 값을 인식하여 그 값이 미리 지정된 프로토콜 필드 값과 일치하면 인터럽트를 발생하는 방법이다.
- <165>      이러한 두 방법에 대해서는 제 1 실시예에 따른 인터럽트 발생 방법(400)과 제 3 실시예에 따른 인터럽트 발생 방법(700) 및 제 5 실시예에 따른 인터럽트 발생 방법(900)에서 상세히 설명했으므로 상세한 설명은 생략한다.
- <166>      도 12는 다른 기술적 과제를 달성하기 위한 본 발명의 제 1 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.



- <167> 도 12를 참조하면, 본 발명의 제 1 실시예에 따른 네트워크 인터페이스 카드(1200)는 제 1 지연 시간 측정 회로(1220), 제 2 지연 시간 측정 회로(1230), 데이터 프레임 카운팅 회로(1240), 판단 회로(1250) 및 인터럽트 발생 회로(1260)를 구비한다.
- <168> 제 1 지연 시간 측정 회로(1220)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 제 1 지연 시간의 측정을 정지한다.
- <169> 제 2 지연 시간 측정 회로(1230)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 2 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 제 2 지연 시간의 측정을 정지한다. 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 제 1 지연 시간보다 짧다.
- <170> 데이터 프레임 카운팅 회로(1240)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임(DFB)의 개수를 카운팅 하고 인터럽트(INTS)에 응답하여 데이터 프레임의 카운팅을 정지한다.
- <171> 판단 회로(1250)는 상기 제 1 지연 시간 측정 회로(1220)의 출력 신호(OTDS1)에 응답하여 제 1 지연 시간 측정 회로(1220)가 제 1 지연 시간에 도달하였는지를 판단하며, 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)에 응답하여 제 2 지연 시간 측정 회로(1230)가 제 2 지연 시간에 도달하였는지를 판단하고, 데이터 프레임 카운팅 회로(1240)의 출력 신호에 응답하여 수신되는 데이터 프레임(DFB)의 개수가 N 개인지를 판단하여, 인터럽트(INTS)의 발생을 제어하는 인터럽트 제어 신호(CTRLS)를 발생한다.

<172> 인터럽트 발생회로(1260)는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.

<173> 네트워크 인터페이스 카드(Network Interface Card)(1200)는 데이터 프레임(DFA)을 수신하여 제 1 지연 시간 측정 회로(1220), 제 2 지연 시간 측정 회로(1230) 및 데이터 프레임 카운팅 회로(1240)로 인가하는 수신 회로(1210) 및 인터럽트(INTS)를 수신하여 전송하는 전송 회로(1270)를 더 구비할 수 있다.

<174> 이하, 도 12를 참조하여 네트워크 인터페이스 카드(1200)의 동작을 살펴본다.

<175> 도 12의 제 1 실시예(1200)는 도 4의 실시예의 인터럽트 발생 방법(400)에 대응된다. 제 1 지연 시간 측정 회로(1220)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 1 지연 시간을 측정한다. 데이터 프레임(DFB)을 수신한 후 바로 인터럽트(INTS)를 발생하는 것이 아니고 제 1 지연 시간이 경과된 후 인터럽트(INTS)를 발생한다. 따라서 처음 데이터 프레임(DFB)이 수신된 후 인터럽트(INTS)가 발생되기 전까지 수신되는 모든 데이터 프레임(DFB)이 한번의 인터럽트(INTS)에 의하여 처리될 수 있다. 따라서, 지연 시간은 하나의 데이터 프레임(DFB)을 수신하는 시간보다는 길어야 한다. 제 1 지연 시간 측정 회로(1220)는 데이터 프레임(DFB)에 응답하여 일정한 시간을 측정하는 타이머로 구성될 수 있다. 그러나 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 타이머에 한정되는 것은 아니며 타이머와 유사한 기능을 하는 다른 장치도 가능하다는 점을 이해할 수 있을 것이다. 제 1 지연 시간 측정 회로(1220)는 인터럽트(INTS)가 발생되면 지연 시간의 측정을 정지한다.

<176> 판단 회로(1250)는 제 1 지연 시간 측정 회로(1220)의 출력 신호(OTDS1)에 응답하여 제 1 지연 시간 측정 회로(1220)가 지연 시간에 도달하였는지를 판단한다. 예를

들어, 제 1 지연 시간 측정 회로(1220)가 미리 정하여진 지연 시간에 도달되지 못하였으면 출력 신호(OTDS1)가 제 1 논리 레벨로 발생되고 지연 시간에 도달되었으면 출력 신호(OTDS1)가 제 2 논리 레벨로 발생되도록 한다. 그러면, 판단 회로(1250)는 출력 신호(OTDS1)의 논리 레벨에 따라 제 1 지연 시간 측정 회로(1220)가 지연 시간에 도달했는지를 판단할 수 있다. 판단 회로(1250)는 또한 출력 신호(OTDS1)의 논리 레벨에 의하여 지연 시간에 도달되지 않은 것으로 판단되면 인터럽트 제어 신호(CTRLS)를 제 1 논리 레벨로 발생하고 출력 신호(OTDS1)의 논리 레벨에 의하여 지연 시간에 도달된 것으로 판단되면 인터럽트 제어 신호(CTRLS)를 제 2 논리 레벨로 발생한다. 그러면 인터럽트 발생 회로(1060)는 인터럽트 제어 신호(CTRLS)의 논리 레벨에 응답하여 인터럽트(INTS)를 발생할 수 있다. 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 상술한 바와 같은 기능을 하도록 판단 회로(1250)나 인터럽트 발생 회로(1260)의 회로 구성을 할 수 있으므로 상세한 내부 회로에 대한 설명은 생략한다.

<177> 또한, 상술한 제 1 및 제 2 논리 레벨은 그 기능이 서로 바뀔 수 있음은 당연하다.

<178> 제 1 지연 시간 측정 회로(1220)가 측정하는 지연 시간이 너무 길게 설정된 경우, 제 1 지연 시간에 도달되어 인터럽트(INTS)가 발생되기 전에 수신된 데이터 프레임(DFB)이 메모리에 가득 차게 되어 더 이상 데이터 프레임(DFB)을 수신할 수 없게 되는 경우가 발생될 수 있다. 따라서 데이터 프레임 카운팅 회로(1240)는 수신되는 데이터 프레임(DFB)의 개수가 N 개인지를 카운팅 한다.

<179> 데이터 프레임(DFB)의 개수가 N 개인지 여부는 데이터 프레임 카운팅 회로(1240)의 출력 신호(ODFS)의 논리 레벨에 의하여 판단할 수 있다. 즉, 수신되는 데이터 프레임

(DFB)의 수가 N 개가되기 전에는 데이터 프레임 카운팅 회로(1240)의 출력 신호(ODFS)는 제 1 레벨로 발생되고, 수신되는 데이터 프레임(DFB)의 수가 N 개가되면 데이터 프레임 카운팅 회로(1240)의 출력 신호(ODFS)는 제 2 레벨로 발생되도록 한다. 그러면 판단 회로(1250)는 데이터 프레임 카운팅 회로(1240)의 출력 신호(ODFS)의 논리 레벨에 따라서 인터럽트 제어 신호(CTRLS)의 논리 레벨도 제 1 논리 레벨이나 제 2 논리 레벨로 발생한다. 인터럽트 발생회로(1260)는 수신되는 데이터 프레임(DFB)의 개수가 N 개에 도달한 경우에 대응되는 논리 레벨을 가지는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.

<180> 데이터 프레임 카운팅 회로(1240)는 수신되는 데이터 프레임(DFB)의 개수를 세는 카운터로 구성될 수 있다. 그러나 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 카운터에 한정되는 것은 아니며 카운터와 유사한 기능을 하는 다른 장치도 가능하다는 점을 이해할 수 있을 것이다.

<181> 수신되는 데이터 프레임(DFB) 사이의 간격이 큰 경우, 제 1 지연 시간의 길이가 길게 설정된다면 첫 번째 데이터 프레임(DFB)이 처리될 때까지 걸리는 시간이 상대적으로 길게 된다. 이러한 문제를 해결하기 위해서, 데이터 프레임(DFB)사이의 수신 시간이 일정 값을 넘어간다면 인터럽트(INTS)를 발생시켜 이미 수신된 데이터 프레임(DFB)을 처리하는 것이다.

<182> 즉, 제 2 지연 시간 측정 회로(1230)는 데이터 프레임이 수신된 후 다음 데이터 프레임(DFB)이 수신될 때까지의 시간을 측정한다. 그리고 판단 회로(1050)는 제 1 지연 시간 측정 회로(1220) 및 데이터 프레임 카운팅 회로(1240)의 출력 신호(OTDS1, ODFS) 이외에도 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)를 수신하고, 수신되는 데이

터 프레임 사이의 시간 간격이 미리 정하여진 제 2 지연 시간에 도달되었는지를 판단한다. 인터럽트 발생 회로(1260)는 판단 회로(1250)에서 출력되는 인터럽트 제어 신호(CTRLS)를 수신하여 인터럽트(INTS)를 발생한다.

<183> 제 2 지연 시간에 도달되었는지의 여부는, 제 1 지연 시간 측정 회로(1220)의 동작과 유사하게 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)의 논리 레벨에 의하여 판단할 수 있다. 즉, 데이터 프레임(DFB)사이의 시간 간격이 제 2 지연 시간에 도달되기 전에는 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)는 제 1 레벨로 발생되고, 수신되는 데이터 프레임(DFB)사이의 시간 간격이 제 2 지연 시간에 도달되면 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)는 제 2 레벨로 발생되도록 한다. 그러면 판단 회로(1250)는 제 2 지연 시간 측정 회로(1230)의 출력 신호(OTDS2)의 논리 레벨에 따라서 인터럽트 제어 신호(CTRLS)의 논리 레벨을 제 1 논리 레벨이나 제 2 논리 레벨로 발생한다. 인터럽트 발생 회로(1260)는 수신되는 데이터 프레임 사이의 시간 간격이 제 2 지연 시간에 도달된 경우에 대응되는 논리 레벨을 가지는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.

<184> 제 2 지연 시간 측정 회로(1230)는 제 1 지연 시간 측정 회로(1220)와 마찬가지로 타이머로 구성될 수 있다. 그러나 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 타이머에 한정되는 것은 아니며 타이머와 유사한 기능을 하는 다른 장치도 가능하다는 점을 이해할 수 있을 것이다.

<185> 네트워크 인터페이스 카드(Network Interface Card)(1200)는 데이터 프레임(DFA)을 수신하여 제 1 및 제 2 지연 시간 측정 회로(1220, 1230), 데이터 프레임 카운팅 회로(1240)로 데이터 프레임(DFB)을 인가하는 수신 회로(1210)와 인터럽트 발생 회로(1260)에

서 발생된 인터럽트(INTS)를 수신하여 상위 프로토콜로 전송하는 전송 회로(1270)를 더 구비한다. 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 상술한 바와 같은 기능을 하도록 수신 회로(1210)나 전송 회로(1270)의 회로 구성을 할 수 있으므로 상세한 내부 회로에 대한 설명은 생략한다.

<186> 또한 도 12의 실시예에 따른 네트워크 인터페이스 카드(1200)의 동작은 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 도 4의 인터럽트 발생 방법(400)으로부터 이해할 수 있으므로 상세한 설명은 생략된다.

<187> 도 13은 다른 기술적 과제를 달성하기 위한 본 발명의 제 2 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.

<188> 도 13의 네트워크 인터페이스 카드(1300)는 제 1 지연 시간 측정 회로(1320), 데이터 프레임 카운팅 회로(1340), 타입 인식 회로(1330), 판단 회로(1350) 및 인터럽트 발생 회로(1360)를 구비한다.

<189> 제 1 지연 시간 측정 회로(1320)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 지연 시간의 측정을 정지한다. 데이터 프레임 카운팅 회로(1340)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임(DFB)의 갯수를 카운팅하고 인터럽트(INTS)에 응답하여 데이터 프레임(DFB)의 카운팅을 정지한다.

<190> 타입 인식 회로(1330)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임의 타입 필드(type field)의 값을 인식하고 인터럽트(INTS)에 응답하여 타입 필드의 값의 인식을 정지한다.

- <191> 판단 회로(1350)는 제 1 지연 시간 측정 회로(1320)의 출력 신호(OTDS1)에 응답하여 제 1 지연 시간 측정 회로(1320)가 지연 시간에 도달하였는지를 판단하며, 데이터 프레임 카운팅 회로(1340)의 출력 신호(ODFS)에 응답하여 수신되는 데이터 프레임(DFB)의 개수가 N 개인지를 판단하고 타이밍 인식 회로(1330)의 출력 신호(OPRS)에 응답하여 인식한 타이밍 값과 미리 지정한 타이밍 값이 일치하는지를 판단하여, 인터럽트(INTS)의 발생을 제어하는 인터럽트 제어 신호(CTRLS)를 발생한다.
- <192> 인터럽트 발생 회로(1360)는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.
- <193> 도 13을 참조하면, 도 13의 실시예에 따른 네트워크 인터페이스 카드(1300)는 도 12의 네트워크 인터페이스 카드(1200)의 구성에서 제 2 지연 시간 측정 회로(1230)대신 타이밍 인식 회로(1330)를 더 구비한다. 즉, 제 1 지연 시간 측정 회로(1320) 및 데이터 프레임 카운팅 회로(1340)를 이용하는 동작은 도 12의 네트워크 인터페이스 카드(1200)와 동일하며 추가적으로 데이터 프레임(DFB)의 타이밍 값을 인식하는 기능이 추가된 것이다.
- <194> 또한 네트워크 인터페이스 카드(1300)의 동작은 도 7의 인터럽트 발생 방법(700)에 대응된다.
- <195> 수신되는 데이터 프레임들 중에는 정해진 시간 내에 반드시 처리되어야 하는 데이터를 포함하는 데이터 프레임들이 있다. 따라서 이러한 데이터 프레임들을 미리 지정하고 이러한 데이터 프레임들이 수신되는 경우 이들 데이터 프레임들을 인식하여 인터럽트를 발생한다.



- <196> 데이터 프레임에는 목적 주소(destination address), 소스 주소(source address), 타입(type), 데이터(data), CRC(cyclic redundancy check)등을 저장하는 필드(field)들이 존재한다. 그 중 타입(type)필드에는 데이터 프레임에 저장되는 데이터의 타입이 어떠한 것인지에 대한 정보가 저장된다.
- <197> 일정한 시간 내에 반드시 처리되어야 하는 데이터의 타입을 미리 지정하고, 수신되는 데이터 프레임의 타입 필드를 검색하여 검색된 타입 값이 미리 지정한 타입 값과 동일하면 인터럽트를 발생한다.
- <198> 데이터 프레임에서 타입 필드의 위치는 정해져 있다. 보통, 목적 주소(destination address)와 소스 주소(source address)가 12 바이트(byte)의 길이를 가진다. 그리고 타입 필드는 2 바이트 길이를 가진다.
- <199> 따라서 타입 인식 회로(1330)는 수신되는 데이터 프레임중 앞에서 12 바이트 다음의 2 바이트를 읽어 타입 필드의 값을 인식한다. 인식된 타입 필드의 값과 미리 지정된 타입 값이 일치하면 타입 인식 회로(1330)는 출력 신호(OPRS)를 제 1 논리 레벨로 발생한다. 인식된 타입 필드의 값과 미리 지정된 타입 값이 일치하지 않으면 타입 인식 회로(1330)는 출력 신호(OPRS)를 제 2 논리 레벨로 발생한다. 여기서, 제 1 논리 레벨은 하이 레벨, 제 2 논리 레벨은 로우 레벨일 수 있다. 그러나 반대의 경우도 가능하다.
- <200> 타입 인식 회로는 수신되는 데이터 프레임에서 타입 필드의 위치를 인식하고, 타입 필드에 저장된 값과 미리 지정된 타입 값이 일치되는 지 여부를 판단한다. 타입 인식 회로는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이다.

- <201> 도 13의 실시예에 따른 네트워크 인터페이스 카드(1300)는 수신 회로(1310)와 전송 회로(1370)를 더 구비할 수 있다. 수신 회로(1310)와 전송 회로(1370)의 기능은 도 12의 실시예에서 설명된 바와 동일하다.
- <202> 도 13의 실시예에 따른 네트워크 인터페이스 카드(1300)의 동작은 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 도 7의 인터럽트 발생 방법(700) 으로부터 이해할 수 있으므로 상세한 설명은 생략된다.
- <203> 도 14는 다른 기술적 과제를 달성하기 위한 본 발명의 제 3 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.
- <204> 도 14의 네트워크 인터페이스 카드(1400)는 제 1 지연 시간 측정 회로(1420), 데이터 프레임 카운팅 회로(1440), 프로토콜 인식 회로(1430), 판단 회로(1450) 및 인터럽트 발생 회로(1460)를 구비한다.
- <205> 제 1 지연 시간 측정 회로(1420)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 지연 시간의 측정을 정지한다. 데이터 프레임 카운팅 회로(1440)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임(DFB)의 갯수를 카운팅 하고 인터럽트(INTS)에 응답하여 데이터 프레임(DFB)의 카운팅을 정지한다.
- <206> 프로토콜 인식 회로(1430)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임의 데이터 필드 내부에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 인터럽트(INTS)에 응답하여 프로토콜 필드의 값의 인식을 정지한다.

- <207> 판단 회로(1450)는 제 1 지연 시간 측정 회로(1420)의 출력 신호(OTDS1)에 응답하여 제 1 지연 시간 측정 회로(1420)가 지연 시간에 도달하였는지를 판단하며, 데이터 프레임 카운팅 회로(1440)의 출력 신호(ODFS)에 응답하여 수신되는 데이터 프레임(DFB)의 개수가 N 개인지를 판단하고 프로토콜 인식 회로(1430)의 출력 신호(OPPS)에 응답하여 인식한 프로토콜 필드의 값과 미리 지정한 프로토콜 필드의 값이 일치하는지를 판단하여, 인터럽트(INTS)의 발생을 제어하는 인터럽트 제어 신호(CTRLS)를 발생한다.
- <208> 인터럽트 발생 회로(1460)는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.
- <209> 도 14를 참조하면, 도 14의 실시예에 따른 네트워크 인터페이스 카드(1400)는 도 12의 네트워크 인터페이스 카드(1200)의 구성에서 제 2 지연 시간 측정 회로(1230)대신 프로토콜 인식 회로(1430)를 구비한다. 즉, 제 1 지연 시간 측정 회로(1420) 및 데이터 프레임 카운팅 회로(1440)를 이용하는 동작은 도 12의 네트워크 인터페이스 카드(1200)와 동일하며 추가적으로 데이터 프레임(DFB)의 프로토콜 필드의 값을 인식하는 기능이 추가된 것이다.
- <210> 또한 네트워크 인터페이스 카드(1400)의 동작은 도 9의 인터럽트 발생 방법(900)에 대응된다.
- <211> 도 9의 인터럽트 발생 방법(900)에서 설명된 것과 같이 데이터 프레임의 데이터 필드에는 이더넷(ethernet)보다 상위 계층에 존재하는 프로토콜에서 생성된 패킷이 위치한다. 인터넷 프로토콜(IP : Internet Protocol ;이하 IP 라고 한다.)이 이더넷(ethernet)보다 상위 계층에 존재하는 프로토콜 중 가장 많이 사용되는 프로토콜이다.

- <212> IP 패킷에는 IP 패킷 헤더와 IP 패킷 데이터의 두 영역으로 나뉘어 진다. 도 14의 네트워크 인터페이스 카드(1400)의 프로토콜 인식 회로(1430)는 IP 패킷 헤더의 프로토콜 필드의 값을 인식한다. 프로토콜 필드에는 IP 패킷 데이터에 저장된 프로토콜 데이터의 종류를 나타내는 제어 정보가 저장되어 있다.
- <213> 프로토콜 인식 회로(1430)는 수신되는 데이터 프레임에 IP 패킷이 존재하는지 여부를 판단하고, IP 패킷이 존재하는 경우 프로토콜 필드의 값을 추출하여 인식한다. 또한 미리 지정된 프로토콜 필드의 값과 동일한지 여부를 판단하는 기능을 한다. 이러한 기능을 하는 프로토콜 인식 회로(1430)는 본 발명의 기술분야에서 통상의 지식을 가진 자라면 회로 구성이 가능할 것이므로 상세한 설명은 생략한다.
- <214> 인식된 프로토콜 필드의 값과 미리 지정된 프로토콜 필드의 값이 일치하면 프로토콜 인식 회로(1430)는 출력 신호(OPPS)를 제 1 논리 레벨로 발생한다.
- <215> 인식된 프로토콜 필드의 값과 미리 지정된 프로토콜 필드의 값이 일치하지 않으면 프로토콜 인식 회로(1430)는 출력 신호(OPPS)를 제 2 논리 레벨로 발생한다. 여기서, 제 1 논리 레벨은 하이 레벨, 제 2 논리 레벨은 로우 레벨일 수 있다. 그러나 반대의 경우도 가능하다.
- <216> 미리 지정된 프로토콜 필드 값은 긴급히 인터럽트를 발생시킬 필요성에 의하여 사용자가 지정한다.
- <217> 판단 회로(1450)는 프로토콜 인식 회로(1430)의 출력 신호(OPPS)의 논리 레벨에 따라서 인터럽트 제어 신호(CTRLS)의 논리 레벨을 제 1 논리 레벨이나 제 2 논리 레벨로 발생한다.

- <218>        인터럽트 발생회로(1260)는 수신되는 데이터 프레임의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 경우에 대응되는 논리 레벨을 가지는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.
- <219>        인터럽트(INTS)가 발생되면 제 1 지연 시간 측정 회로(1420), 프로토콜 인식회로(1430) 및 데이터 프레임 카운팅 회로(1440)의 동작은 새로운 데이터 프레임이 수신될 때까지 정지된다.
- <220>        도 14의 실시예에 따른 네트워크 인터페이스 카드(1400)는 수신 회로(1410)와 전송 회로(1470)를 더 구비할 수 있다. 수신 회로(1410)와 전송 회로(1470)의 기능은 도 12의 실시예에서 설명된 바와 동일하다.
- <221>        도 14의 실시예에 따른 네트워크 인터페이스 카드(1400)의 동작은 본 발명의 기술 분야에서 통상의 지식을 가진 자라면 도 9의 인터럽트 발생 방법(900)으로 부터 이해할 수 있으므로 상세한 설명은 생략된다.
- <222>        도 15는 다른 기술적 과제를 달성하기 위한 본 발명의 제 4 실시예에 따른 네트워크 인터페이스 카드를 나타내는 블록도이다.
- <223>        도 15의 네트워크 인터페이스 카드(1500)는 제 1 지연 시간 측정 회로(1520); 제 2 지연 시간 측정 회로(1530), 데이터 프레임 카운팅 회로(1550), 타입 인식 회로(1540), 프로토콜 인식 회로(1555), 판단 회로(1560) 및 인터럽트 발생회로(1570)를 구비한다.
- <224>        제 1 지연 시간 측정 회로(1520)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 지연 시간의 측정을 정지한다.

- <225> 제 2 지연 시간 측정 회로(1530)는 수신되는 데이터 프레임(DFB)에 응답하여 일정한 제 2 지연 시간을 측정하고 인터럽트(INTS)에 응답하여 제 2 지연 시간의 측정을 정지한다. 제 2 지연 시간은 수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 제 1 지연 시간보다 짧다.
- <226> 데이터 프레임 카운팅 회로(1550)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임(DFB)의 개수를 카운팅 하고 인터럽트(INTS)에 응답하여 데이터 프레임(DFB)의 카운팅을 정지한다.
- <227> 타입 인식 회로(1540)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임의 타입 필드(type field)의 값을 인식하고 인터럽트(INTS)에 응답하여 타입 필드의 값의 인식을 정지한다.
- <228> 프로토콜 인식 회로(1555)는 수신되는 데이터 프레임(DFB)에 응답하여 데이터 프레임(DFB)의 데이터 필드에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 인터럽트(INTA)에 응답하여 프로토콜 필드 값의 인식을 정지한다.
- <229> 판단 회로(1560)는 제 1 지연 시간 측정 회로(1520)의 출력 신호(OTDS1)에 응답하여 제 1 지연 시간 측정 회로(1520)가 제 1 지연 시간에 도달하였는지를 판단하며, 제 2 지연 시간 측정 회로(1530)의 출력 신호(OTDS2)에 응답하여 제 2 지연 시간 측정 회로(1530)가 제 2 지연 시간에 도달하였는지를 판단하며, 데이터 프레임 카운팅 회로(1550)의 출력 신호(ODFS)에 응답하여 수신되는 데이터 프레임(DFB)의 개수가 N 개인지를 판단하고 타입 인식 회로(1540)의 출력 신호(OPRS)에 응답하여 인식한 타입 값과 미리 지정된 타입 값이 일치하는 지를 판단하고, 상기 프로토콜 인식 회로의 출력 신호에 응답하여 상기 프로토콜 인식 회로가 인식한 프로토콜 값이 미리 지정된 프로토콜 값과 일치하

는 지를 판단하여, 인터럽트(INTS)의 발생을 제어하는 인터럽트 제어 신호(CTRLS)를 발생한다.

<230> 인터럽트 발생 회로(1570)는 인터럽트 제어 신호(CTRLS)에 응답하여 인터럽트(INTS)를 발생한다.

<231> 도 15를 참조하면, 도 15의 실시예에 따른 네트워크 인터페이스 카드(1500)는 도 12, 13 및 14의 네트워크 인터페이스 카드(1200, 1300, 1400)의 구성을 합하여 놓은 것과 같다.

<232> 네트워크 인터페이스 카드(1500)의 동작은 도 11의 인터럽트 발생 방법(1100)에 대응되므로 동작에 대한 상세한 설명은 생략된다.

<233> 이상에서와 같이 도면과 명세서에서 최적 실시예가 개시되었다. 여기서 특정한 용어들이 사용되었으나, 이는 단지 본 발명을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 발명의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 기술분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

#### 【발명의 효과】

<234> 상술한 바와 같이 본 발명에 따른 인터럽트 발생 방법 및 네트워크 인터페이스 카드는 발생하는 인터럽트의 수를 최소화하면서도, 한 번의 인터럽트 발생으로 많은 수의 데이터 프레임을 처리할 수 있는 장점이 있다.



**【특허청구범위】****【청구항 1】**

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

- (a) 데이터 프레임을 수신하는 단계 ;
- (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고 상기 수신되는 데이터 프레임의 갯수를 카운팅하는 단계 ;
- (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 ;
- (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개 인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 ;
- (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간에 도달하였으면 인터럽트를 발생하고 상기 제 2 지연 시간에 도달하지 않았으면 상기 (b) 단계로 돌아가는 단계 ;
- (f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간 및 상기 제 2 지연 시간을 측정하는 동작 및 상기 데이터 프레임의 개수를 카운팅하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및
- (g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 2】**

제 1항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 3】**

제 1항에 있어서, 상기 제 2 지연 시간은,

수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧은 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 4】**

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

- (a) 데이터 프레임을 수신하는 단계 ;
- (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 패킷 지연 시간을 측정하는 단계 ;
- (c) 상기 패킷 지연 시간에 도달했는지를 판단하고, 상기 패킷 지연 시간에 도달했으면 인터럽트를 발생하고 상기 패킷 지연 시간에 도달하지 않았으면 상기 (b) 단계로 돌아가는 단계 ;
- (e) 상기 발생된 인터럽트에 응답하여 상기 패킷 지연 시간을 측정하는 동작을 중지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(f) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 5】**

제 4항에 있어서, 상기 패킷 지연 시간은,  
수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값인 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 6】**

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

- (a) 데이터 프레임을 수신하는 단계 ;
- (b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 갯수를 카운팅하며, 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하는 단계 ;
- (c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 ;
- (d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개 인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 ;
- (e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고, 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 (b) 단계로 돌아가는 단계 ;

(f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅하는 동작 및 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 7】**

제 6항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 8】**

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

(a) 데이터 프레임을 수신하는 단계 ;

(b) 상기 수신되는 데이터 프레임의 타입 필드(type field)의 값을 인식하는 단계 ;

(c) 상기 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b)단계로 돌아가는 단계 ;

(d) 상기 발생된 인터럽트에 응답하여 상기 타입 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(e) 새로운 데이터 프레임을 수신하고 상기(b)단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

【청구항 9】

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

(a) 데이터 프레임을 수신하는 단계 ;

(b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 갯수를 카운팅 하며, 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계 ;

(c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 ;

(d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 ;

(e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 지를 판단하고, 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고 일치하지 않으면 상기 (b) 단계로 돌아가는 단계 ;

(f) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅 하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(g) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 10】**

제 9항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 11】**

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

(a) 데이터 프레임을 수신하는 단계 ;

(b) 상기 수신되는 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계 ;

(c) 상기 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는 지를 판단하고 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b)단계로 돌아가는 단계 ;

(d) 상기 발생된 인터럽트에 응답하여 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(e) 새로운 데이터 프레임을 수신하고 상기(b)단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

【청구항 12】

데이터를 송수신하는 네트워크 인터페이스 카드(Network Interface Card)의 인터럽트 발생 방법에 있어서,

(a) 데이터 프레임을 수신하는 단계 ;

(b) 상기 수신되는 데이터 프레임에 응답하여 미리 정하여진 제 1 지연 시간 및 제 2 지연 시간을 측정하고, 상기 수신되는 데이터 프레임의 갯수를 카운팅 하고 상기 수신되는 데이터 프레임의 타입 필드(type field)의 값을 인식하며 상기 수신되는 데이터 프레임의 데이터 필드에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하는 단계 ;

(c) 상기 제 1 지연 시간에 도달했는지를 판단하고, 상기 제 1 지연 시간에 도달했으면 인터럽트를 발생하는 단계 ;

(d) 상기 제 1 지연 시간에 도달하지 않았으면 상기 데이터 프레임의 개수가 N 개 인지를 판단하고, 상기 데이터 프레임의 개수가 N 개이면 인터럽트를 발생하는 단계 ;

(e) 상기 데이터 프레임의 개수가 N 개가 아니면, 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간에 도달하였으면 인터럽트를 발생하는 단계 ;

(f) 상기 제 2 지연 시간에 도달하지 않았으면, 상기 타입 필드의 값이 미리 지정된 타입 값과 일치하는 지를 판단하고 상기 미리 지정된 타입 값과 일치하면 인터럽트를 발생하는 단계 ;



(g) 상기 미리 지정된 타입 값과 일치하지 않으면, 상기 수신되는 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값이 미리 지정된 프로토콜 필드의 값과 일치하는지를 판단하고 상기 미리 지정된 프로토콜 필드의 값과 일치하면 인터럽트를 발생하고, 일치하지 않으면 상기 (b)단계로 돌아가는 단계 ;

(h) 상기 발생된 인터럽트에 응답하여 상기 제 1 지연 시간 및 상기 제 2 지연 시간을 측정하는 동작, 상기 데이터 프레임의 개수를 카운팅하는 동작, 상기 타입 필드의 값을 인식하는 동작 및 상기 프로토콜 필드의 값을 인식하는 동작을 정지시키고 상기 수신된 데이터 프레임을 전송하는 단계 ; 및

(i) 새로운 데이터 프레임을 수신하고 상기 (b) 단계로 돌아가는 단계를 구비하는 것을 특징으로 하는 인터럽트 발생 방법.

#### 【청구항 13】

제 12항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트 발생 방법.

#### 【청구항 14】

제 12항에 있어서, 상기 제 2 지연 시간은,

수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧은 것을 특징으로 하는 인터럽트 발생 방법.

**【청구항 15】**

수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지하는 제 1 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 일정한 제 2 지연 시간을 측정하고 상기 인터럽트에 응답하여 상기 제 2 지연 시간의 측정을 정지하는 제 2 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지하는 데이터 프레임 카운팅 회로 ;

상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 제 2 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 2 지연 시간 측정 회로가 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생하는 판단 회로 ; 및

상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생하는 인터럽트 발생 회로를 구비하는 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

**【청구항 16】**

제 15항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

【청구항 17】

제 15항에 있어서, 상기 제 2 지연 시간은,

수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧은 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

【청구항 18】

수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지하는 제 1 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하고 상기 인터럽트에 응답하여 상기 타입 필드의 값의 인식을 정지하는 타입 인식 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지하는 데이터 프레임 카운팅 회로 ;

상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 타입 인식 회로의 출력 신호에 응답하여 상기 타입 인식 회로가 인식한 타입 값이 미리 지정된 타입 값과 일

치하는 지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생하는 판단 회로 ; 및

상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생하는 인터럽트 발생 회로를 구비하는 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

【청구항 19】

제 18항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

【청구항 20】

수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지하는 제 1 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 상기 인터럽트에 응답하여 상기 프로토콜 필드 값의 인식을 정지하는 프로토콜 인식 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지하는 데이터 프레임 카운팅 회로 ;

상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하며, 상기 프로토콜 인식 회로의 출력 신호에 응답하여 상기 프로토콜 인식 회로가 인식한 프로토콜 값이 미리 지정된 프로토콜 값과 일치하는 지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생하는 판단 회로 ; 및

상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생하는 인터럽트 발생 회로를 구비하는 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

#### 【청구항 21】

제 18항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

#### 【청구항 22】

수신되는 데이터 프레임에 응답하여 일정한 제 1 지연 시간을 측정하고 인터럽트에 응답하여 상기 제 1 지연 시간의 측정을 정지하는 제 1 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 일정한 제 2 지연 시간을 측정하고 상기 인터럽트에 응답하여 상기 제 2 지연 시간의 측정을 정지하는 제 2 지연 시간 측정 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 타입 필드(type field)의 값을 인식하고 상기 인터럽트에 응답하여 상기 타입 필드의 값의 인식을 정지하는 타입 인식 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 갯수를 카운팅 하고 상기 인터럽트에 응답하여 상기 데이터 프레임의 카운팅을 정지하는 데이터 프레임 카운팅 회로 ;

상기 수신되는 데이터 프레임에 응답하여 상기 데이터 프레임의 데이터 필드(data field)에 존재하는 패킷 헤더의 프로토콜 필드의 값을 인식하고 상기 인터럽트에 응답하여 상기 프로토콜 필드 값의 인식을 정지하는 프로토콜 인식 회로 ;

상기 제 1 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 1 지연 시간 측정 회로가 상기 제 1 지연 시간에 도달하였는지를 판단하고, 상기 제 2 지연 시간 측정 회로의 출력 신호에 응답하여 상기 제 2 지연 시간 측정 회로가 상기 제 2 지연 시간에 도달하였는지를 판단하고, 상기 타입 인식 회로의 출력 신호에 응답하여 상기 타입 인식 회로가 인식한 타입 값이 미리 지정된 타입 값과 일치하는 지를 판단하고, 상기 데이터 프레임 카운팅 회로의 출력 신호에 응답하여 상기 수신되는 데이터 프레임의 개수가 N 개인지를 판단하며, 상기 프로토콜 인식 회로의 출력 신호에 응답하여 상기 프로토콜 인식 회로가 인식한 프로토콜 값이 미리 지정된 프로토콜 값과 일치하는 지를 판단하여, 상기 인터럽트의 발생을 제어하는 인터럽트 제어 신호를 발생하는 판단 회로 ; 및

상기 인터럽트 제어 신호에 응답하여 상기 인터럽트를 발생하는 인터럽트 발생 회로를 구비하는 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

**【청구항 23】**

제 22항에 있어서, 상기 제 1 지연 시간은,

첫 번째 데이터 프레임을 수신한 때부터 측정되는 일정한 시간인 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

**【청구항 24】**

제 22항에 있어서, 상기 제 2 지연 시간은,

수신되는 데이터 프레임들 사이의 시간 간격을 측정한 값으로서, 상기 제 1 지연 시간보다 짧은 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

**【청구항 25】**

제 15항, 제 18항, 제 20항 및 제 22항 중 어느 한 항에 있어서, 상기 네트워크 인터페이스 카드(Network Interface Card)는,

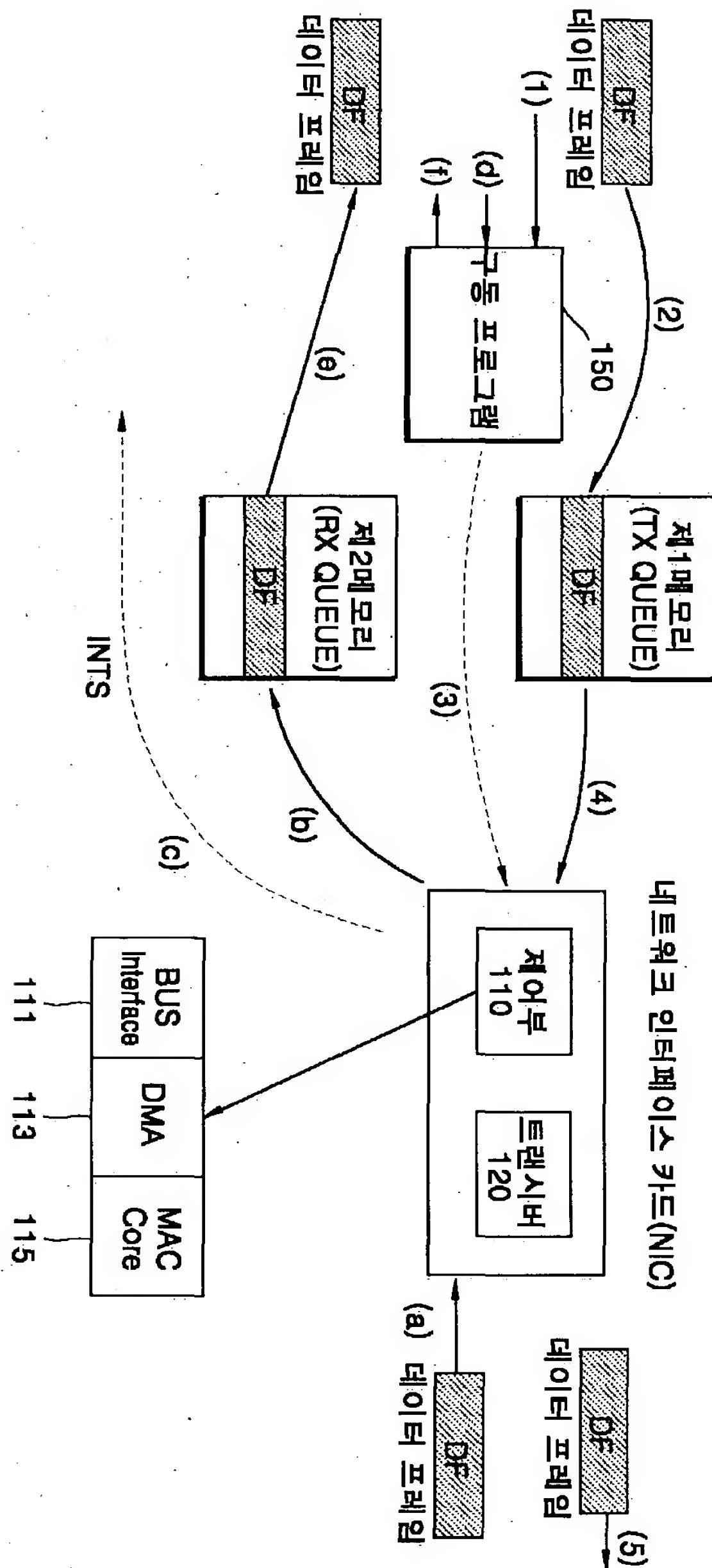
상기 데이터 프레임을 수신하여 상기 제 1 지연 시간 측정 회로, 상기 제 2 지연 시간 측정 회로, 상기 데이터 프레임 카운팅 회로, 상기 타임 인식 회로 및 상기 프로토콜 인식 회로로 인가하는 수신 회로 ; 및

상기 인터럽트를 수신하여 전송하는 전송 회로를 더 구비하는 것을 특징으로 하는 인터럽트의 발생횟수를 최소화하는 네트워크 인터페이스 카드(Network Interface Card).

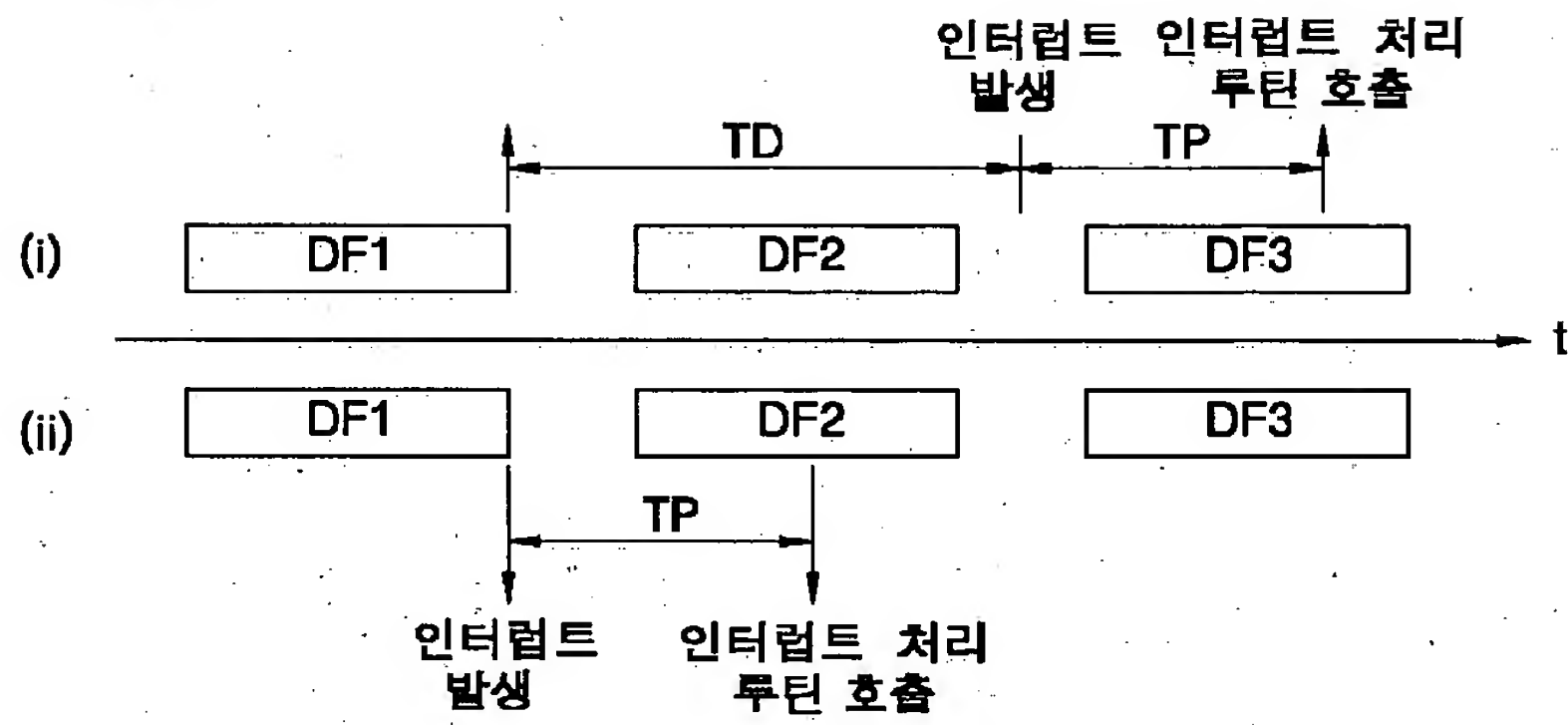


【도면】

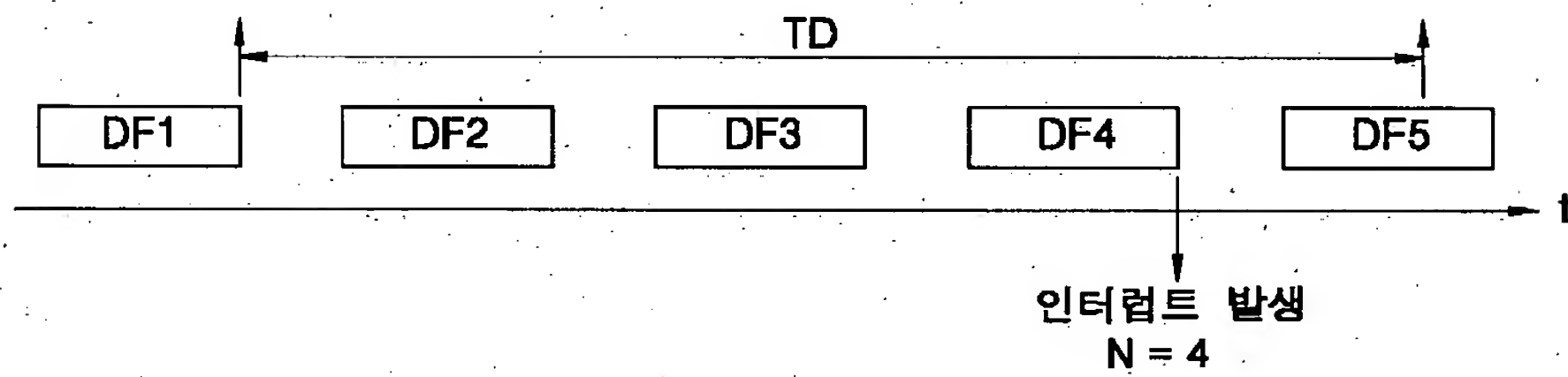
【도 1】



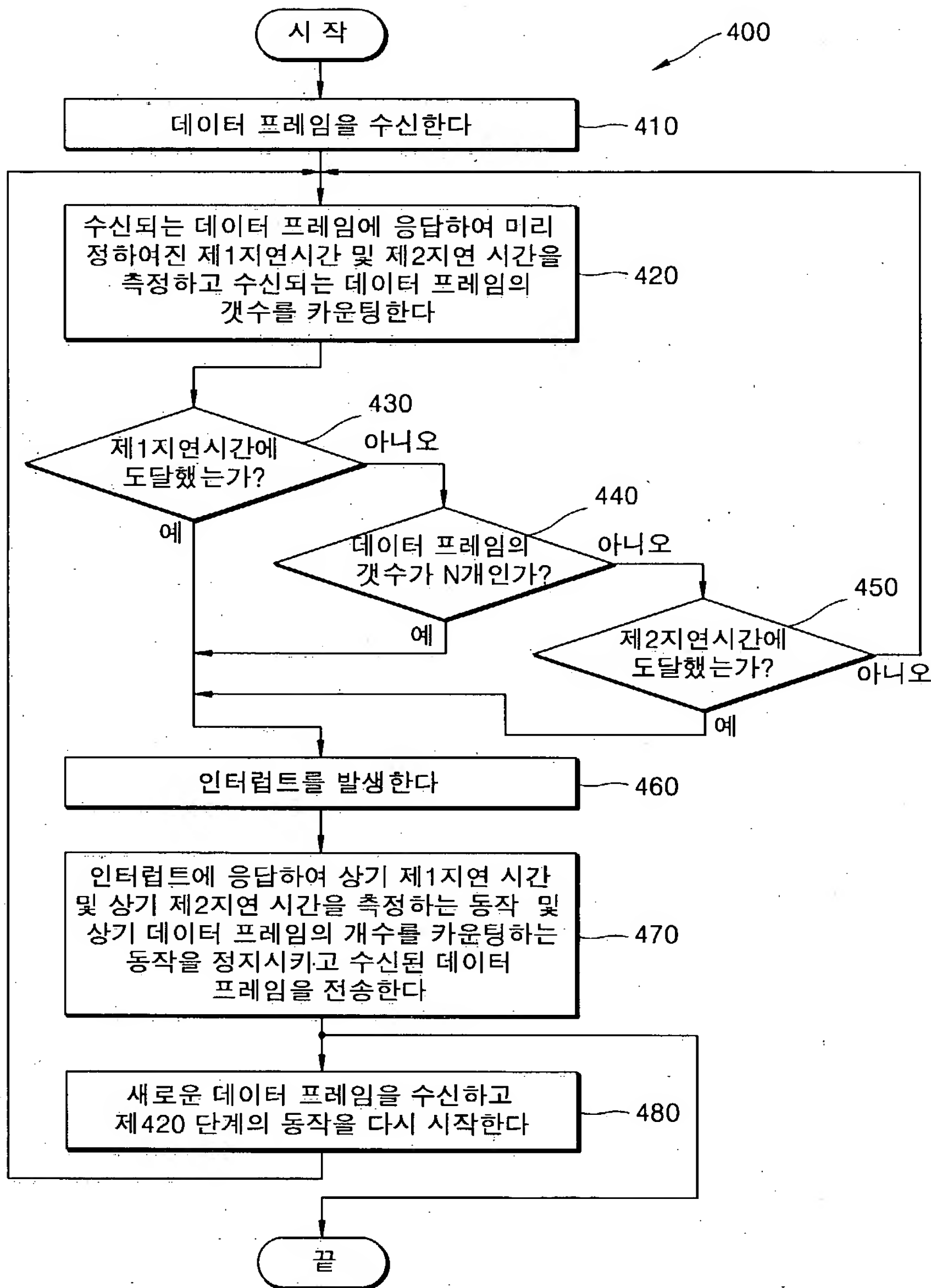
【도 2】



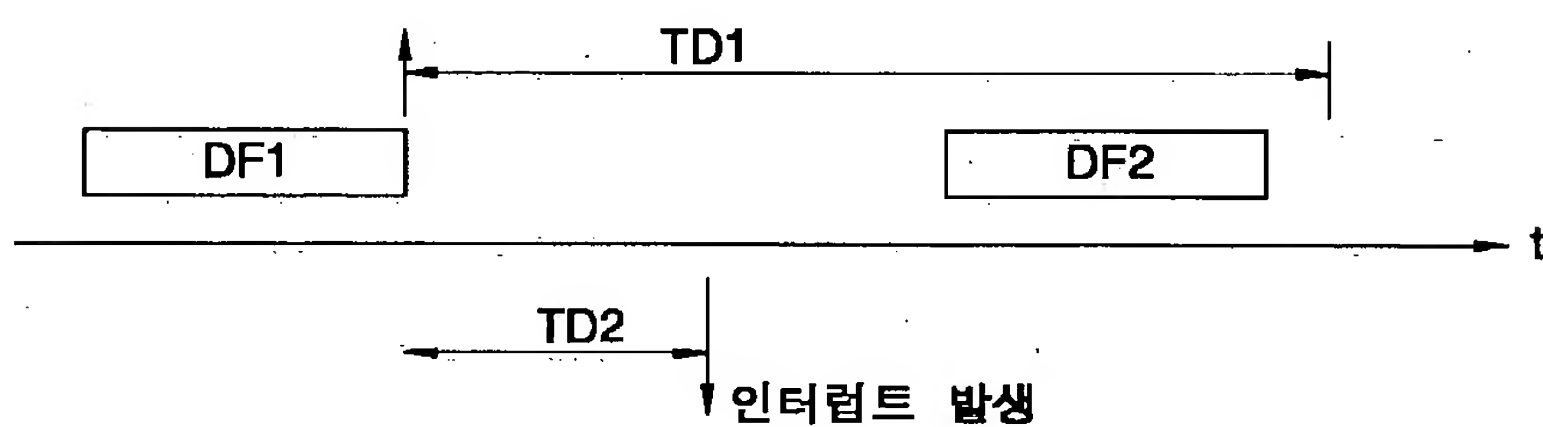
【도 3】



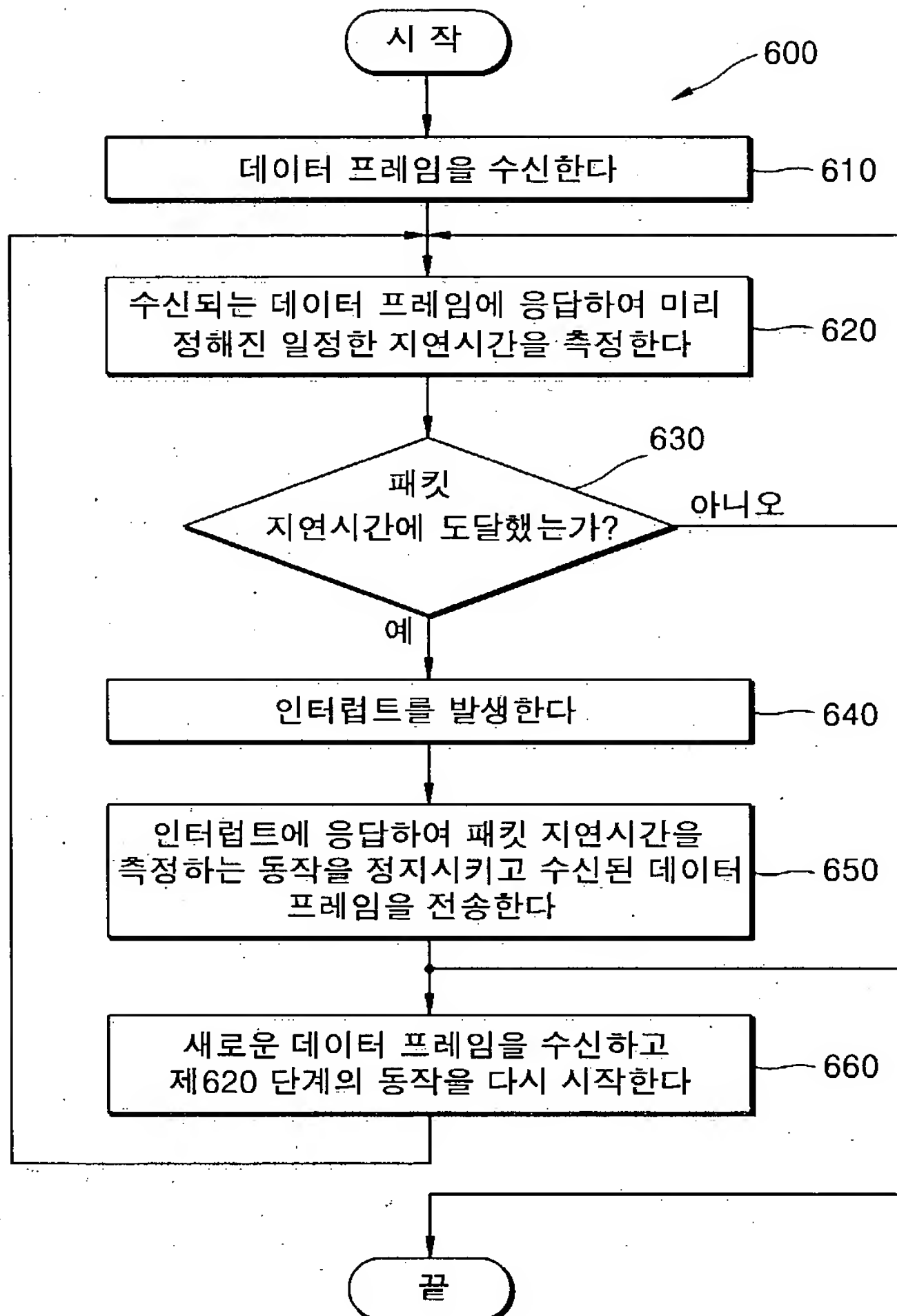
【도 4】



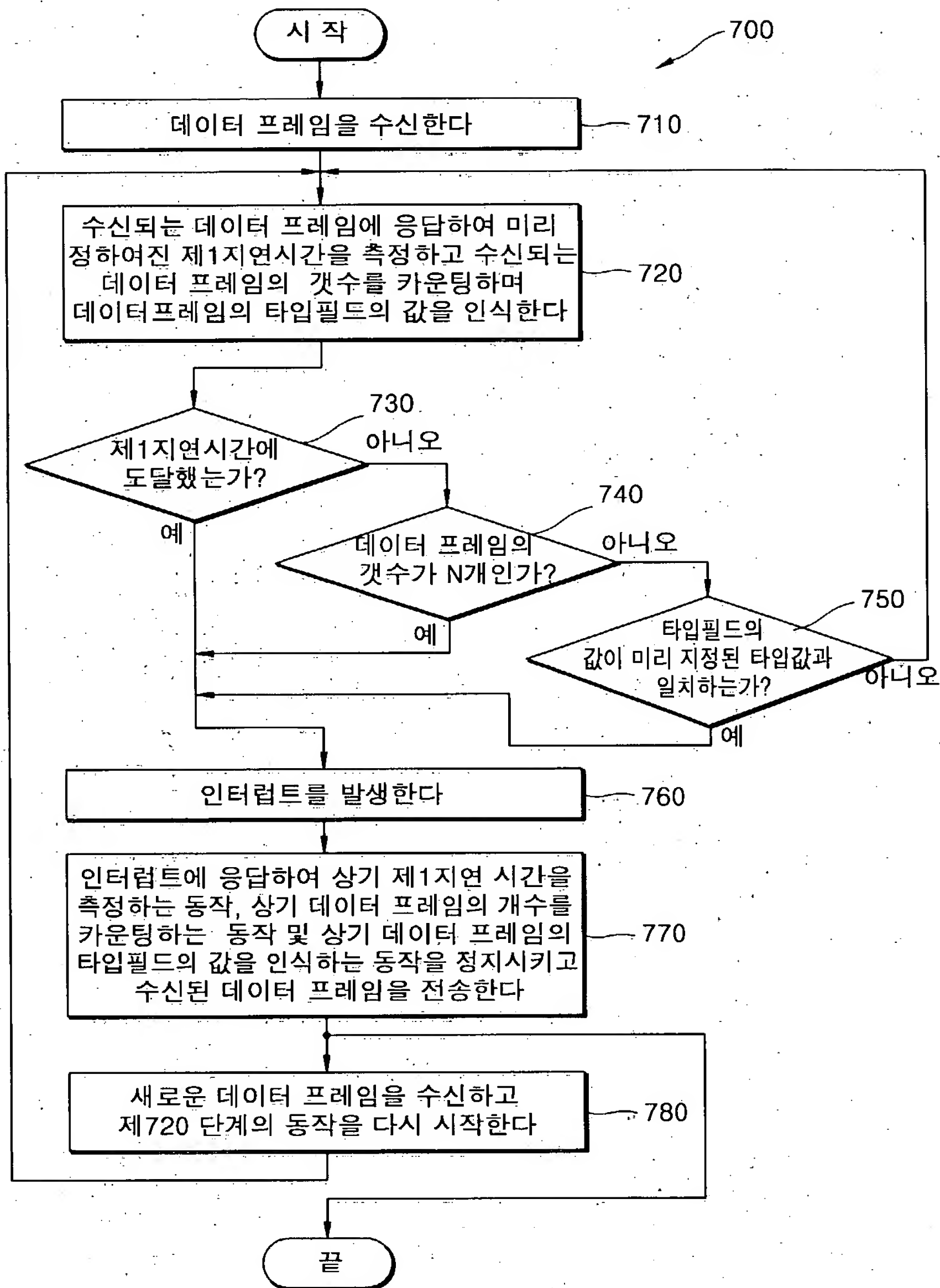
【도 5】



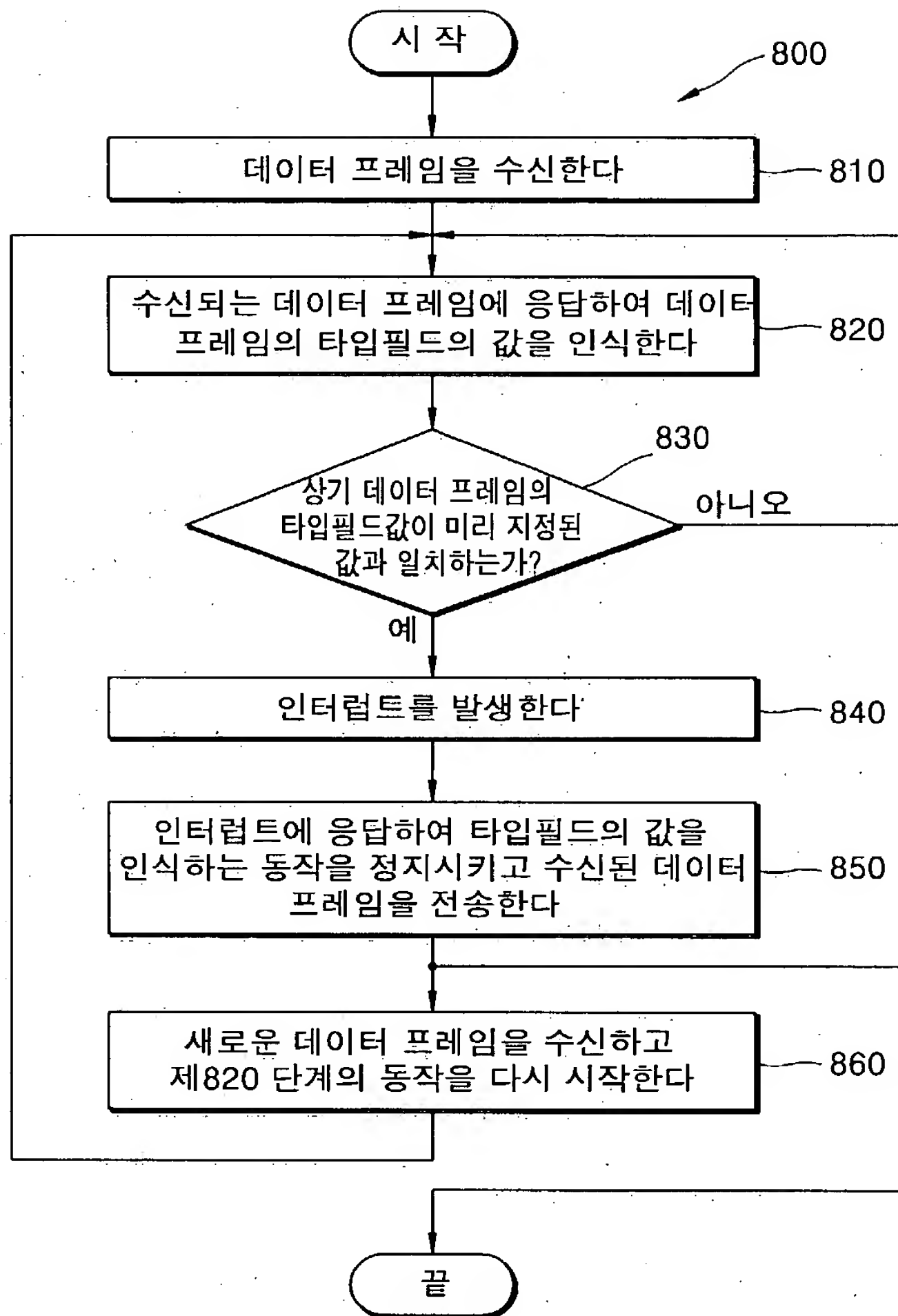
【도 6】



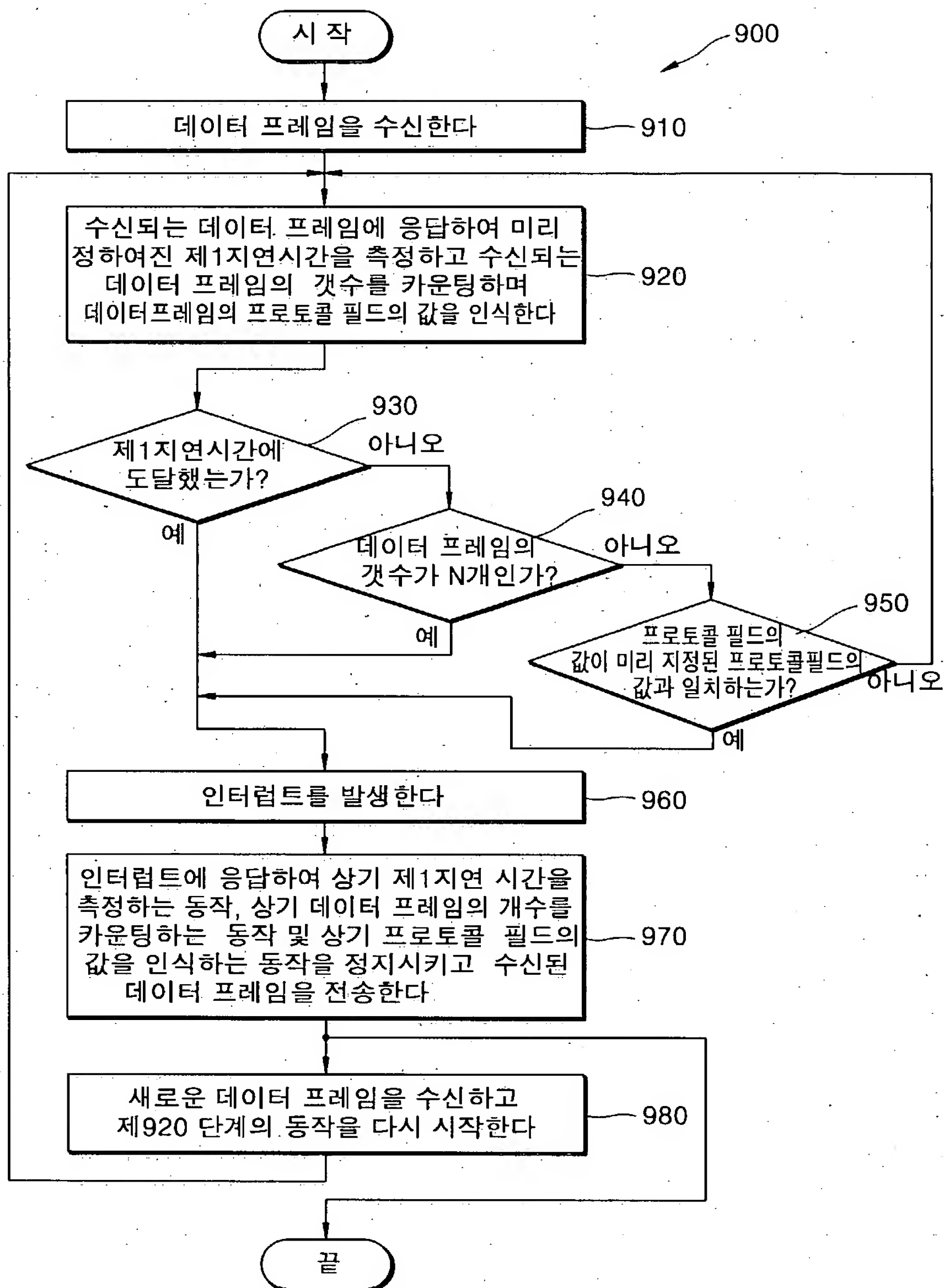
【도 7】



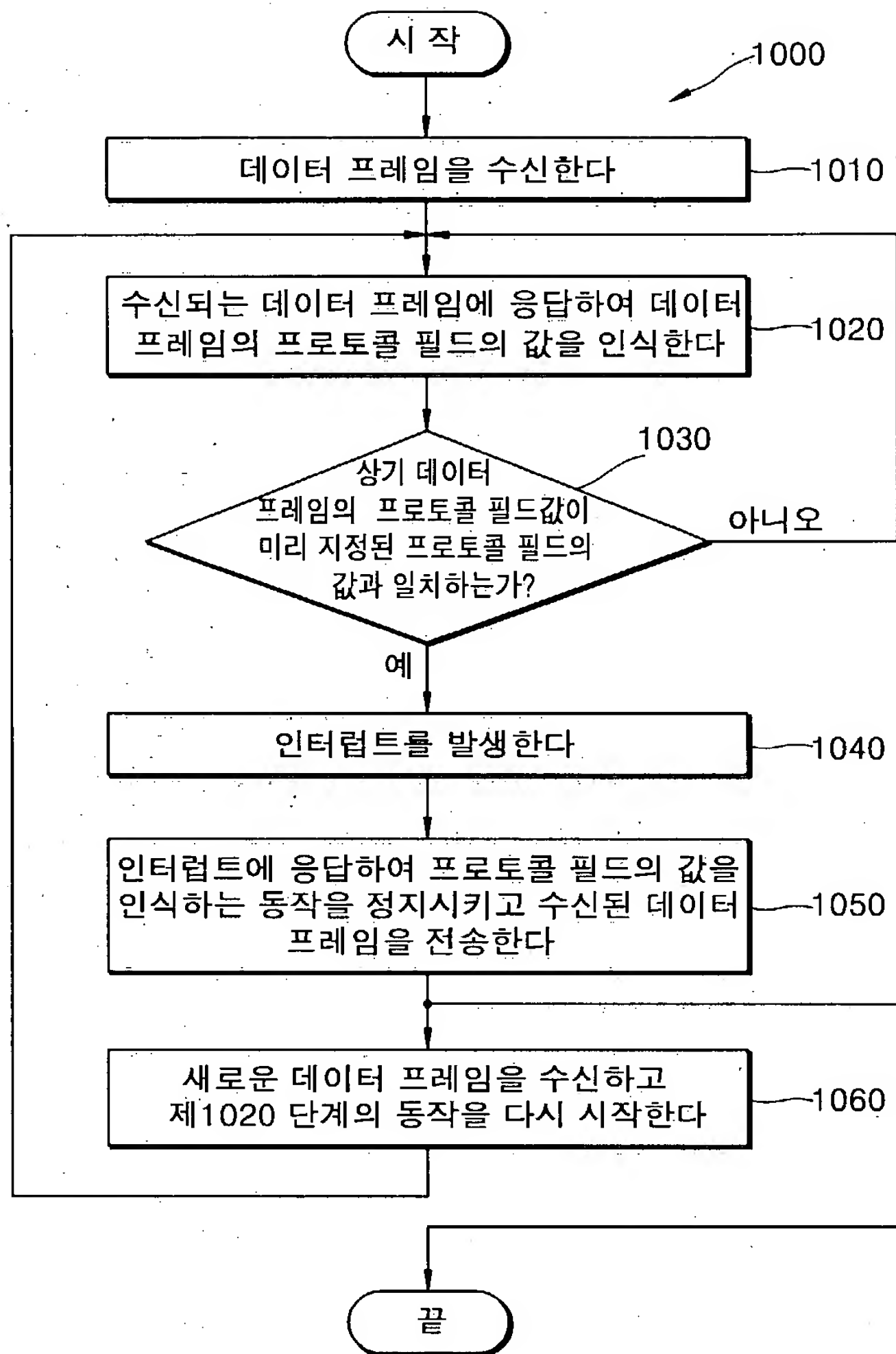
【도 8】



【도 9】

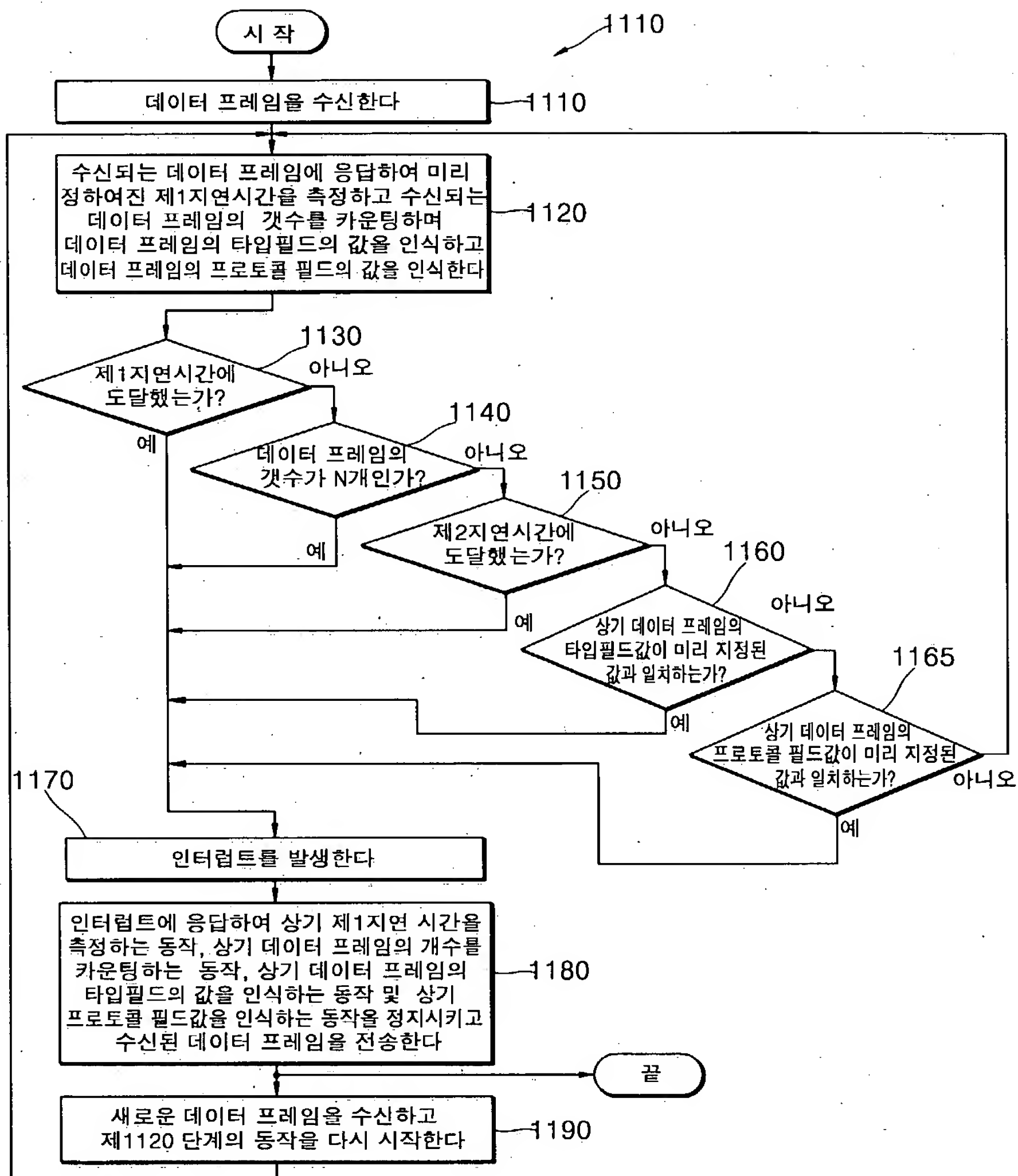


【도 10】

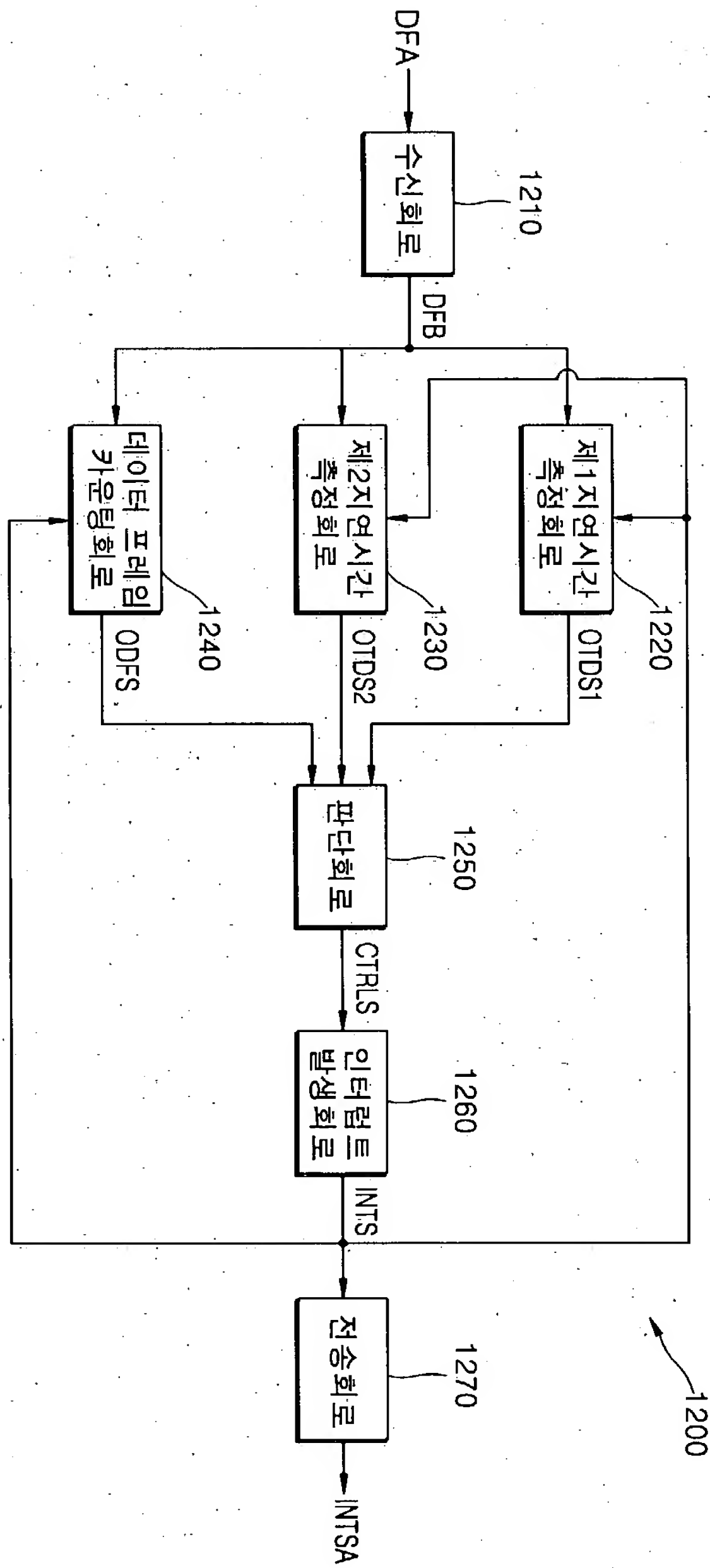




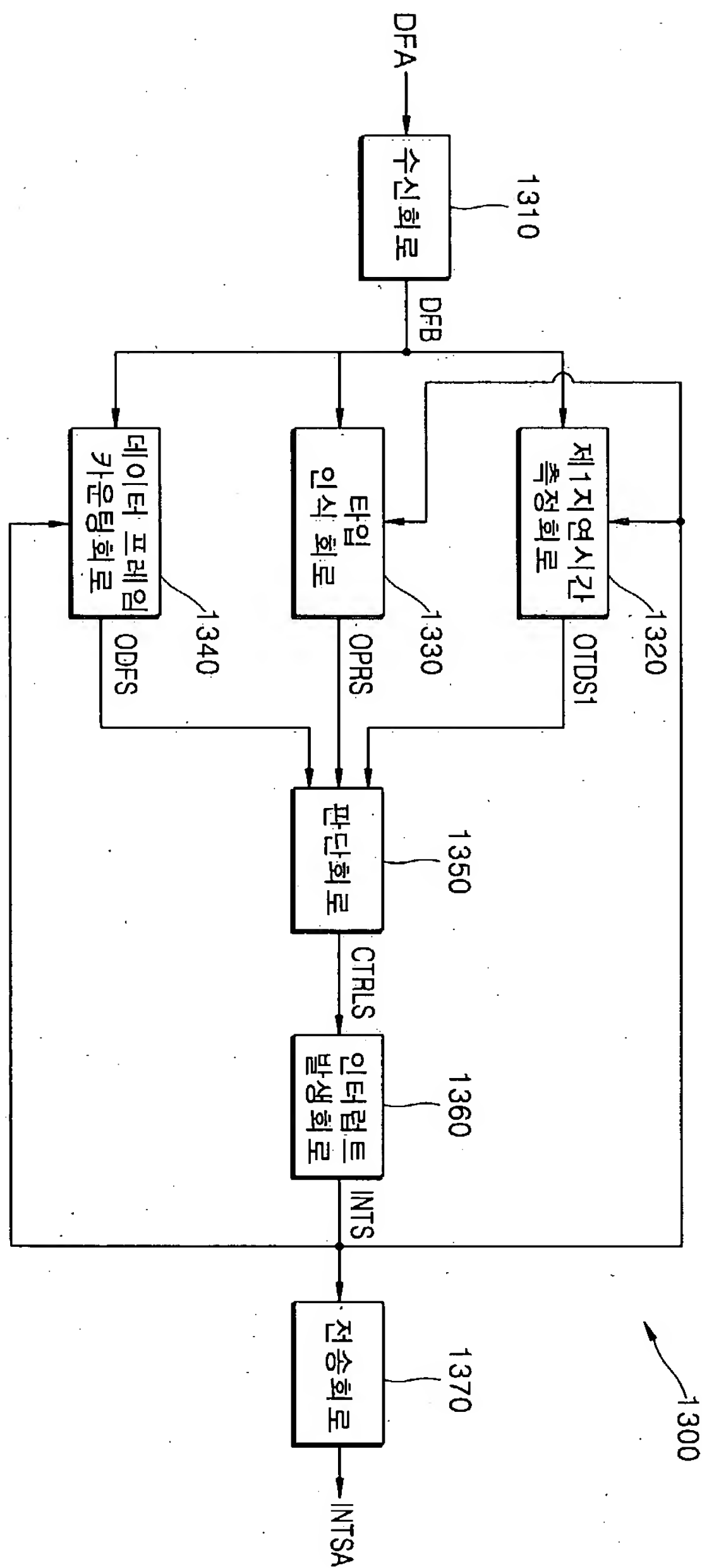
【도 11】



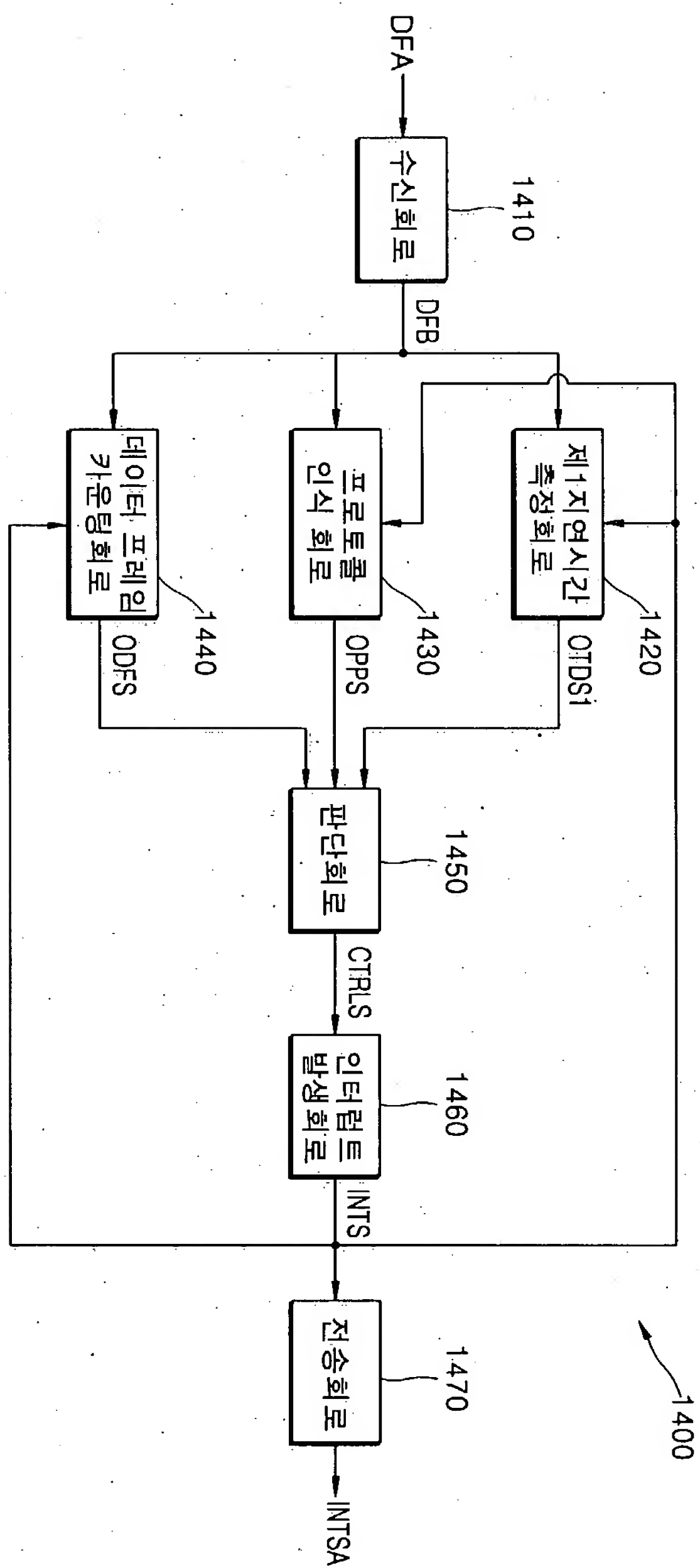
【도 12】



【도 13】



【도 14】



【도 15】

